

**Ismoilov A.A., Yusupov R.A., Boymurodov.,
A.X. Xurramov A.J.**

C++ da dasturlash asoslari

O'quv qo'llanma

Chirchiq-2021 y.

УЎК 004.4(075)

КБК 32.97-018

С 49

**Ismoilov A.A., Yusupov R.A., Boymurodov A.X.
Xurramov A.J. “C++ da dasturlash asoslari”. O‘quv
qo‘llanma. – Chirchiq, TVChDPI, 2021. – 296 b.**

Annotatsiya

O‘quv qo‘llanma dasturlashni mustaqil o‘rganuvchilar, oliy o‘quv yurtlarida ta’lim oluvchi talabalar va o‘qituvchilarga mo‘ljallangan. O‘quv qo‘llanmadagi har bir mavzu nazariy materiallar, amaliy mashg‘ulot bo‘yicha masalalar yechimlari bilan birgalikda mustaqil topshiriqlar bilan to‘ldirilgan.

Taqrizchilar:

Raximov N.O.- t.f.d.dotsent., TATU.

Axmedjonov D.G‘.- t.f.d., TVChDPI,

O‘zbekiston Respublikasi Oliy va o‘rta maxsus ta’lim vazirligi
ning 2021-yil 31 maydagi 237-sonli buyrug‘iga asosan o‘quv
qo‘llanma sifatida nashrga tavsiya etilgan.

ISBN 978-9943-7476-6-1

**© Ismoilov A.A., Yusupov R.A., Boymurodov A.X.
Xurramov A.J. 2021.**

© «Ishonchli hamkor» nashriyoti, 2021.

KIRISH

Ushbu qo'llanma nomutaxassislik yo'nalishi talabalarini fan olimpiadaga tayyorlash uchun mo'ljallangan.

Dasturlashni har kungi izlanishsiz va amaliyot'siz o'rganib bo'lmaydi. Ushbu qo'llanma talabalarga ko'pgina masalalarni yechishda duch keladigan qiyinchiliklarni va qo'l yordamida yechib bo'lmaydigan masalalarni yechishda amaliy yordam beradi.

Dasturlashni endigina boshlayotgan talabalar turgan gapki ayrim masalalarni ishlash jarayonida uning algoritmini va dasturini yechishda bir muncha qiyinchiliklarga duch keladi. Bu muammoni yechish uchun amaliyotga katta e'tibor qaratish lozim. Mazkur qo'llanmada imkon qadar misol sifatida bir qancha murakkab masalalarni yechishga asos bo'la oladigan tipik algoritmlarning yechish usullari ko'rsatib berilgan.

Qo'llanmada to'la variantga qaraganda dasturlashning boshlang'ich kursida o'qish maqsadga muvofiq bo'lgan konstruksiyalar tushib qoldirilgan. Shu bilan bilan birga muallif qo'yilgan masalaga mos tarzda, C++ dasturlash tilining afzalliklarini ko'rsatish uchun tavsifiy vositalarni to'liq qamrab olishga harakat qildi.

Qo'llanmada dasturlashning boshlang'ich kursiga mo'ljallanganini hisobga olib, dasturlash tili maksimal darajada sodda va tushinarli qilib tahlil qilindi va ko'pgina konstruksiyalar misol sifatida masalalar yechishning kompyuterda tekshirib ko'rilgan dasturlari ko'rsatildi.

Amaliyotda keltirilgan ishlar qo'llaniladigan sonli uslublar va tipik masalalarni yechish uchun misollarni o'z ichiga qamrab olgan kirish qismiga ega. Har bir turdagi misol uchun dasturlar keltirilgan. Nomutaxassislik yo'nalishida ta'lim olayotgan talabalarga dars sifatida o'tilayotgan aniq fanlarda keltirilgan misol va masalalardan misollar keltirilgan.

Mualliflar keltirilgan algoritm, dastur va usullarning asliligiga to'liq da'vo qilishmaydi. Masalalarning bir qismi mualliflar tomonidan tuzilgan bo'lib, qolganlari adabiyotlar ro'yxatida keltirilgan dasturlashtirish qo'llanmalari va masalalar to'plamidan tanlab olingan.

“SHAXSIY KOMPYUTER ARXITEKTURASI”. XAVFSIZLIK TEXNIKASI, KOMPYUTER SINFIGA ISHLASH QOIDALARI

Nazariy qism. Kompyuterlarning klassifikatsiyasi.

Shaxsiy EHM terminining kelib chiqishi jahon bozorida AQSH ning IBM firmasida "Personal Computer" nomli kichik EHM ning yaratilishi bilan bog'liq.

Shaxsiy EHM ning dastlabki avlodlari 8-razryadli katta integral sxemali (KIS) mikrosxema, mikroprotsessor komplektlari asosida qurilgandir.

Shaxsiy EHM ning muvaffaqiyatlari uni ishlab chiqarishni kengaytirish va yangi takomillashgan, keng ishlatish imkoniyatlariga ega bo'lgan modellarini yaratishga olib keldi.

Hozirgi paytda SHEHM modellari asosan 16, 32 va 64 razryadli KIS va juda katta integral sxemali (JKIS) komplektlar asosida qurilmoqda. Shaxsiy EHM bir vaqtning o'zida bir shaxsning ishlashiga mo'ljallangan bo'lib, ommaviy ravishda foydalaniladigan va boshqa EHM lardan arzonligi, kichik sig'imliligi, yengilligi, juda ham ishonchli ishlashi, tarkibiy qismlarini o'zgartirish osonligi, rivojlangan muloqot vositalariga egaligi, xotira sig'imi juda katta Kbait tashkil etganligi, kabi belgilar bilan farq qiluvchi axborotlarni qayta ishlash vositasidir.

Kompyuterlarning qurilmalari va ularning ishlash asoslari Axborotlarni kiritish qurilmasi:

- klaviatura – kompyuterga axborotni kiritish va uni boshqarishning yagona asosiy qurilmasi hisoblanadi;
- manipulyatorlar – axborotni kiritish uchun qo'shimcha boshqaruv pulti hisoblanadi. Uning turlari:
 - Joystik – kursorni ekranda to'rt yo'nalishdan biriga ko'chirishni ta'minlaydi;
 - Nurli pero – displey ekranidagi nuqtani ko'rsatish uchun yoki tasvirni yuzaga keltirish uchun qo'llanilishi mumkin;
 - “Sichqon” – turidagi mexanik (sharikli) manipulyatori – asosiy uzeli korpus asosidan chiqib stol sirtiga tegib turadigan shar hisoblanadi;

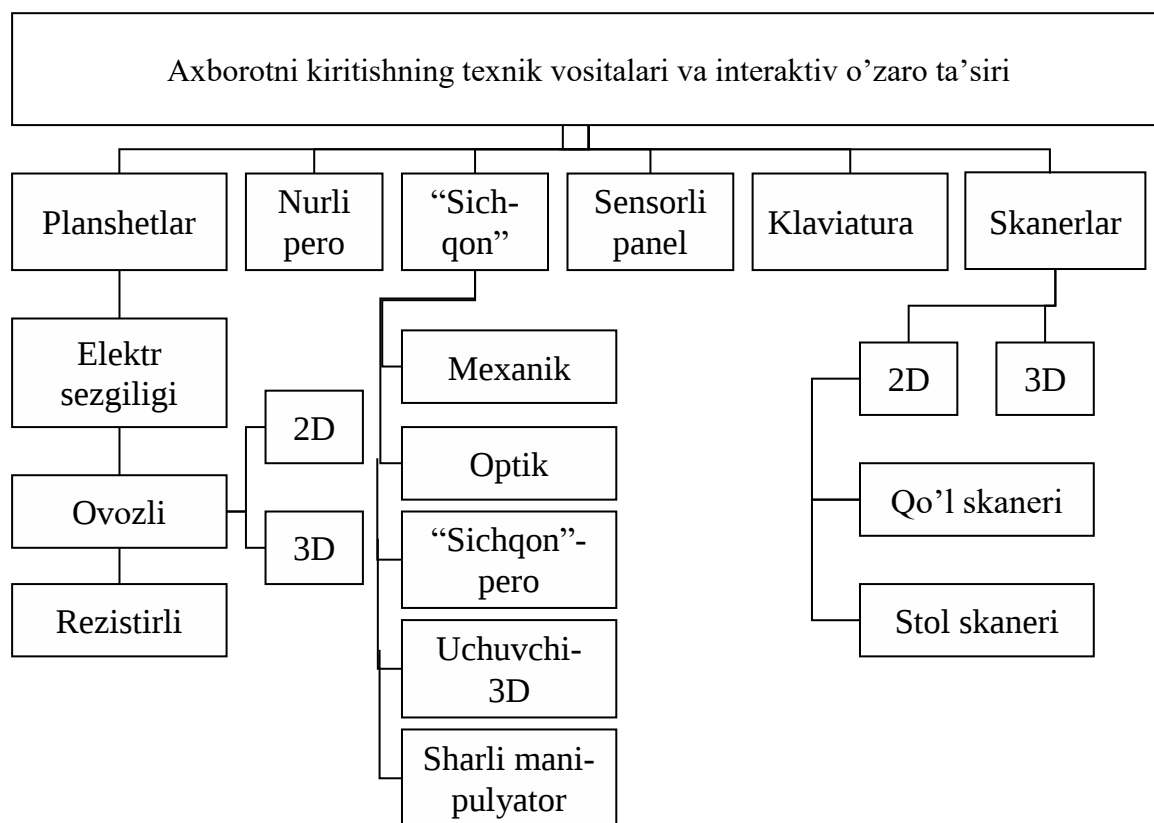
-optik manipulyatorning ishlash prinsipi maxsus optoelektron tarmoq planshetga nurli dioddan nur uzatib va aks etgan signalni qabul qilgan holda manipulyator xarakatini va bosib o‘tilgan masofani qayd etadi;

-Skanerlar – kompyuter xotirasiga tasvir kiritish imkonini beruvchi qurilmaga aytiladi. Skanerlar stol va portativ turlariga ajraladi;

-Grafikli planshet – Skanerdan tayyor tasvirlarni kompyuterga kiritishni ta’minlagan bir vaqtda grafik planshetlar yoki didjitayzerlar (inglizcha digitizer – raqamli uzgartirgich) ular yaratilishini avtomatlashtiradi. Ular kompyuterga chiziqlardan iborat grafik axborotlarni kiritishni ancha osonlashtiradi.

-Sensorli ekranlar – ekranda nuqtani ko‘rsatish ma’nosida keladi;

-Nutqni kiritish vositasi.



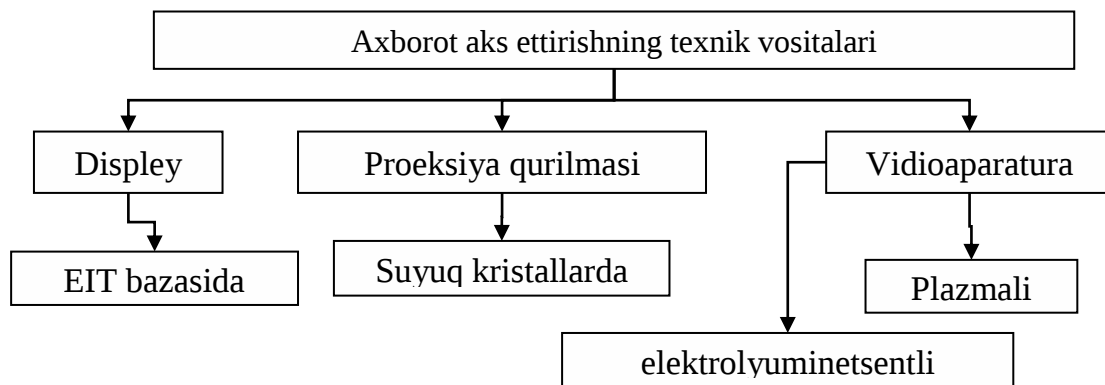
1-rasm. Axborotni kiritishning texnik vositalari.

Axborotlarni chiqarish qurilmalari:

Display va display adapterlari. Display deb matnli va grafik axborotni uzoq vaqt qayd etmasdan aks ettiruvchi qurilmaga aytiladi. Displaylar qo‘yidagi turlarga ajraladi:

- elektron - nurli trubka bazasidagi;
- suyuq kristalli; (anizotrop suyuqligidan iborat).
- plazmali;
- elektrolüminesentli displaylardir.

Vektorli – elektron nur tasvir konturini to‘xtovsiz aniq qilib chizadi. Rastrlı displaylarda esa tasvir nuqtalar matrisasidan hosil bo‘ladi.



2-rasm. Axborotni aks ettirishning texnik vositalari.

Axborotlarni bosib chiqarish qurilmasi:

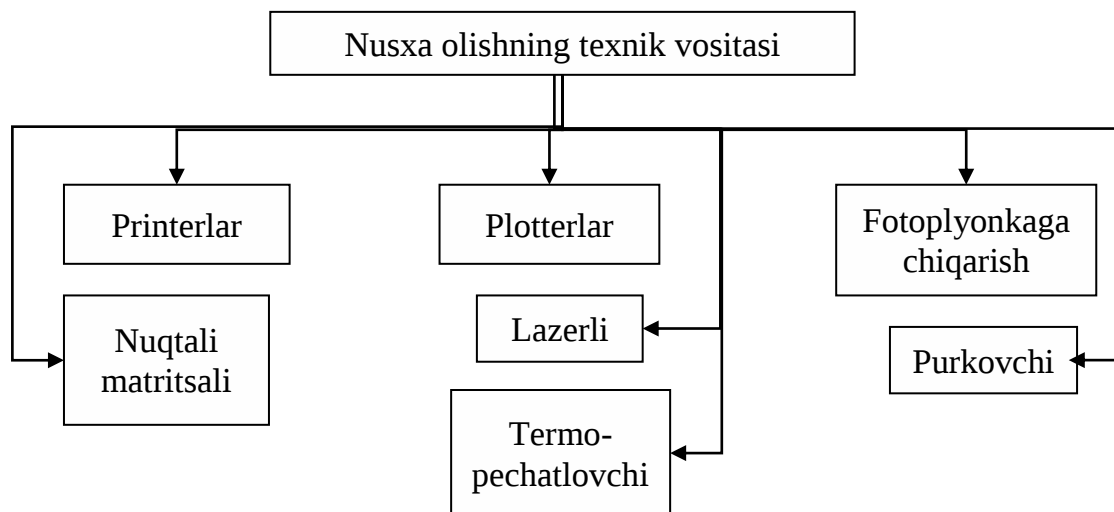
Zarbli printerlar orqali tasvir qog‘ozga mexanik usulda tushiriladi.

Purkovchi printerlar – qog‘ozga tasvirlarni purkash yo‘li bilan tushiradi.

Termografik va matrisa-nuqtali printerlar – ularda nuqtani qo‘yishda qizdiriladigan elektrodlardan foydalanadi.

Elektrofotografik (lazerli) printerlar.

Grafa tuzuvchilar (plotter)- bu rasm yoki jadval shaklidagi ma‘lumotlarni taqdim etuvchi chiqarish qurilmasidir.



3-rasm. Nusxa olishning texnik vositasi

Shaxsiy kompyuterlar foydalanuvchilariga axborot kiritish turli qurilmalarining keng turi tavsiya etiladi. Shunday bo‘lsada, ko‘pincha ShKlar yagona kiritish qurilmasi-klaviatura bilan ishlab chiqariladi. Keng ko‘lamda qo‘llanishi jixatdan klaviaturadan so‘ng uni funksional jixatdan to‘ldiruvchi turli xil manipulyatorlar va asosan “sichqon” xilidagi manipulyatorlar qo‘llaniladi.

ShK-bu qo‘llanilishining hamma bog‘lik va universallik talablarini qondiruvchi stolli yoki ko‘chma EHMdir.

ShKning afzallik tomonlari: narxining arzonligi; atrof-muhit sharoitlariga maxsus talablarsiz foydalanishning avtonomligi; boshqarish, fan, ta‘lim, maishiy turmush, sohasida uning xilma-xil qo‘llanishlariga moslashuvchanligini ta‘minlovchi arxitekturasining tez o‘zgaruvchanligi; ishning yuqori darajada ishonchliligi.

ShK asosiy bloklarining tarkibi va vazifasini ko‘rib chiqamiz.

Mikroprosessor (MP). Bu ShKning markaziy bloki bo‘lib, mashina barcha bloklarning ishini boshqarish va axborot ustidan arifmetik va mantiqiy operatsiyalarni bajarish uchun mo‘ljallangan. Uning tarkibiga quyidagilar kiradi: boshqaruv moslamasi (BM)-mashinaning barcha bloklariga zarur paytda bajariladigan operatsiyalar o‘ziga xos va avvalgi operatsiyalar natijalari bilan o‘zaro bog‘langan boshqaruvning muayyan signallarini uzatadi va shakllantiradi; bajariladigan operatsiyalarda foydalanuvchi xotira uyachasi adresini shakllantiradi va ShKning tegishli bloklariga bu adresni uzatadi.

Arifmetik-mantiqiy moslama (AMM) -son bilan hisoblanadigan va belgili axborotlar ustida barcha arifmetik va mantiqiy operatsiyalarni bajarishga mo'ljallangan; mikroprosessorli xotira (MPX)-mashina ishining yaqin taktlarida hisoblashlarda bevosita foydalaniladigan axborotni qisqa muddatli saqlash, yozib olish va berish uchun xizmat qiladi. MPX registrlarda quriladi va mashinaning yuqori tezligini ta'minlashga xizmat qiladi, chunki asosiy xotira (AX) tez ishlovchi mikroprosessor ishining samaradorligi uchun zarur bo'lgan axborotni yozib olish, izlash va hisoblash tezligini xar doim ham ta'minlay olmaydi. Registrlar-turli uzunlikdagi tez xarakatlanuvchi xotira uyachalaridir; mikroprosessorning interfeys tizimi-ShKning boshqa moslamalar bilan bog'liqligi va aloqasini amalga oshiradi; MP ning ichki interfeys, kirish-chiqish portlari (KChP) boshqarishning buferli yodda tutuvchi registr va chizmalari, tizimli shinani o'z ichiga oladi. Interfeys-kompyuter moslamasining bog'liqlik va aloqa vositalari yig'indisi bo'lib, ularning o'zaro samarali aloqasini ta'minlaydi. Taktli impulslar generatori elektrik impulslar izchilligini yuzaga keltiradi; yuzaga keltirilgan impulslar chastotasi mashinaning taktli chastotasini belgilaydi. Mashina ishining bir takt vaqtini yoki oddiy aytganda mashina ish taktini qo'shni impulslar o'rtasida o'tgan vaqt belgilaydi.

Tizimli shina kompyuterning asosiy interfeys tizimi bo'lib, uning barcha moslamalarida o'zaro bog'lanish va aloqani ta'minlaydi. Tizimli shinani boshqarish mikroprosessor yordamida bevosita yoki ko'pincha qo'shimcha mikro chizma-boshqarishning asosiy signallarini shakllantiruvchi shina kontrollerlari orqali amalga oshadi. Asosiy xotira (AX) mashinaning boshqa bloklari bilan axborotni saqlash va operativ almashish uchun mo'ljallangan. AX ikki turdagi yodda tutuvchi moslama: doimiy va operativ yodda tutuvchi (DYoTM), (OYoTM) moslamalarga ega.

Tashqi xotira ShKning tashqi qurilmalariga oiddir va vazifalarni xal qilish uchun talab qilinishi mumkin bo'lgan istalgan axborotni uzoq saqlash uchun mo'ljallangan. Tashqi xotira moslamalari sifatida kasetali, magnit tasmali yodda tutuvchi

moslamalar (strimmerlar), optik disklardagi yig'uvchilar (CD-ROM) va boshqalar foydalaniladi.

Ta'minlash manbai ShK avtonom va tarmoqli energiya ta'minoti tizimlarini ta'minlovchi blokdir.

Taymer-bu zarur bo'lganda joriy vaqtni avtomatik belgilab beruvchi ichki mashina soatidir.

Tashqi moslama (TM) istalgan hisoblash kompleksining eng muhim tarkibiy qismidir. Shuni aytish kifoyaki, TM ba'zan qiymatiga ko'ra butun ShK qiymatining 50-80 foizini tashkil etadi. TM tarkibi va xususiyatlari bo'yicha ShKni boshqaruv tizimlari va umuman xalq xo'jaligida qo'llash imkoniyati va samaradorligiga ko'p jixatdan bog'liq bo'ladi.

Multimedia vositalari-bu insonga o'zi uchun tabiiy muhit: tovush, video, grafika, matnlar va boshqalardan foydalanib, kompyuter bilan muloqotda bo'lishga imkon beruvchi texnik va dasturiy vositalar majmuidir. Multimedia vositalariga axborotni nutqli kiritish va chiqarish moslamalari; hozirda keng tarqalgan skanerlar; yuqori sifatli video va tovushli platalar, tasvirni videokamera videomagnitafondan ko'chiruvchi va uni ShKga kirituvchi videoegallash (video-grabber) platasi; kuchaytirgichli, tovushli kolonkali, katta videoekranli yuqori sifatli akustik va video aks ettiruvchi tizimlar kiradi. Tizimli shina va ShK MPga namunaviy tashqi moslamalar bilan bir qatorda integral mikrochizmalı ayrim qo'shimcha platalarni ham kiritish mumkin ular mikroprosessor: matematik qo'shimcha prosessor, xotiraga to'g'ri kiradigan kontroller, kiritish/chiqarish qo'shimcha prosessori, uzilishlar kontrolleri va boshqalarning funksional imkoniyatlarini kengaytiradi va yaxshilaydi.

Mikroprosessor, boshqacha aytganda, markaziy prosessor-Central Processing Unit(CPU)-bir yoki bir necha katta (KICh) yoki o'ta katta (UKICh) integral chizmalar ko'rinishida bajarilgan axborotni qayta ishlashning funksional tugallangan dasturiy-boshqaruv moslamasidir.

Mikroprosessor ma'lumotlari shinasining razryadligi umuman ShK razryadligini belgilaydi. MP adresi shinasining razryadligi-uning adresli makonini belgilaydi.

Adresli makon-bevosita mikroprosessorga jo'natilishi mumkin bo'lgan asosiy xotira uyachalarining eng ko'p miqdoridir.

Birinchi mikroprosessor 1971 yilda Intel (AKSh) firmasi tomonidan chiqarildi. Hozirgi paytda bir necha yuz turli xil MPlar chiqarilmoqda. Biroq, eng mashxur va keng tarqalganlari Intel va unga o'xshash firmalar MPlaridir. Barcha MPlar uch guruxga bo'linadi:

-buyruqlarning to'liq to'plami bo'lgan CISC tipidagi MP (Complex Instruction Set Computing);

-buyruqlarning qisqartirilgan to'plami bo'lgan RISC turidagi MP (Reduced Instruction Set Computing) (hozirgi paytda bu modellar ishlab chiqish bosqichida);

IBM PC (International Business Mashine) turidagi zamonaviy ShKlarning ko'pida CISC tipidagi MPdan foydalaniladi.

Mustaqil bajarish uchun topshiriqlar

1. ShK ichki qurilmalari
2. ShK tashqi qurilmalari
3. Xotira turlari
4. Klaviatura va uning funksiyalari
5. Operativ xotira va uning funksiyalari
6. Disk yurituvchilar va ular orasidagi farqlar
7. Kompyuterni ishga tayyorlash va ishlash jarayonidagi

ketma-ketliklar

8. ShKlarning xossalarini va xususiyatlari
9. Tizimli blok va uning turlari
10. Prosessorlar va mikroprosessorlar
11. Ovoz kartalarining va vidio kartalar
12. Printerlar va ularning turlari
13. ShKlarning asosiy xarakteristikasi
14. Operativ ichki xotira, doimiy xotira, magnitli diskda

to'plagich

AXBOROTNI IFODALASH USULLARI. AXBOROT O‘LCHOV BIRLIKLARI

Nazariy qism. Axborotning xususiyatlari va ko‘rinishlari.

Korxona va tashkilotlarda boshqaruv qarorlarini qabul qilish jarayoni ma’lumotlarning katta oqimidagi zarur axborotni ko‘rib chiqish, tahlil etish va oqilona foydalanishni ko‘zda tutadi. Axborot o‘rganish va ularni tanlash ko‘p mehnat talab qiladigan ish bo‘lib qimmat turadigan jarayon hisoblanadi. Shuning uchun uni tasniflash zarur.

Tasnif tizimini axborotga qo‘llash uchun quyidagicha yondashish zarur:

1. axborotni tahlil qilish tasnif va qiyoslash vositalari asosida bo‘lishi kerak;
2. tasnif asosiga turli omil va koordinat tizimlari quyilishi lozim;
3. ayrim omil va koordinat tizimlari u yoki bu farqlarga asoslanishi mumkin, masalan aniq bir bozor va tovarlarni bilish zarur; tahlilning mohiyati va teranligi avvalo berilgan axborot turi va xajmiga bog‘liq.

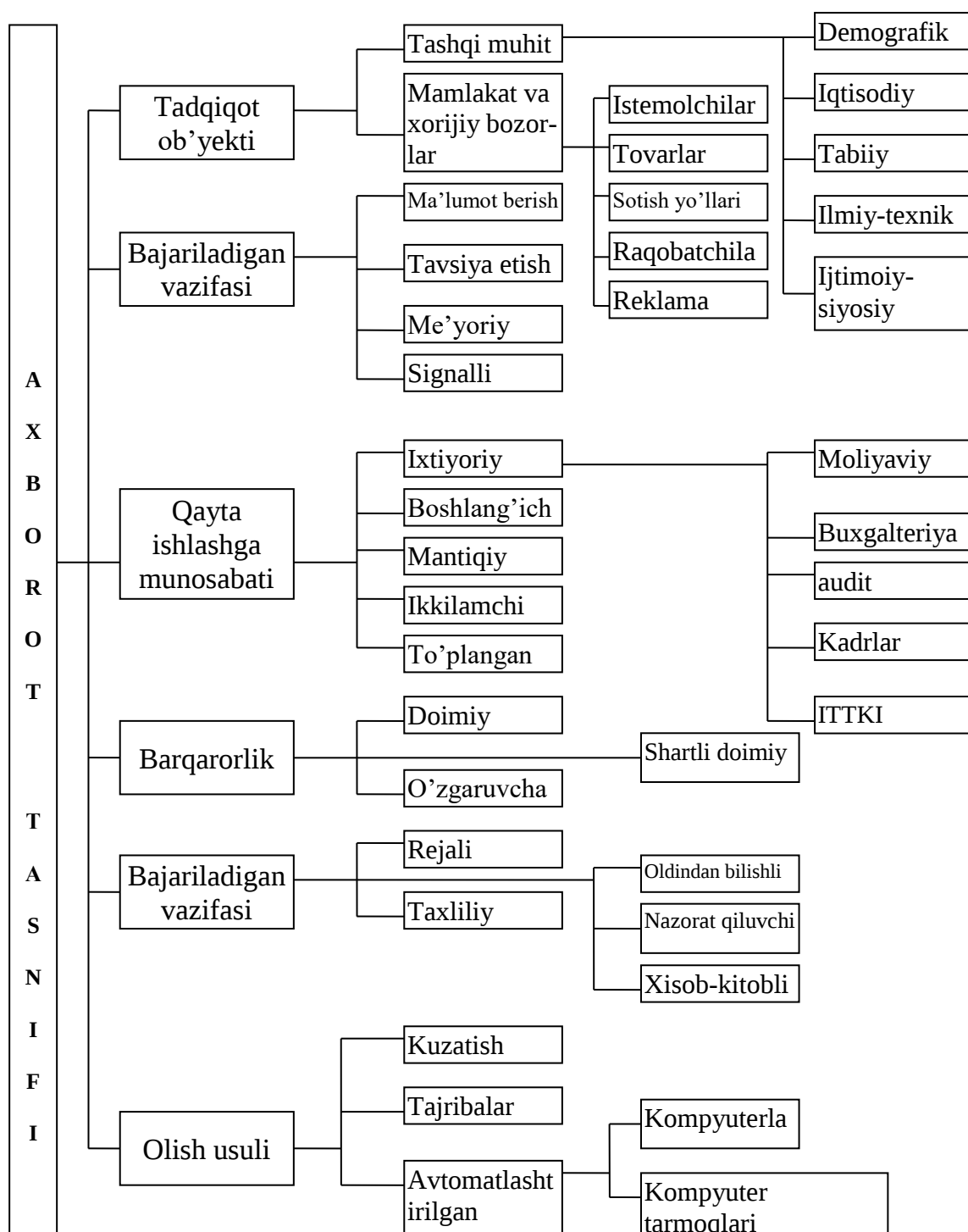
Axborotni har xil belgilarga qarab tasniflash mumkin.

1. **Axborot olish usuli bo‘yicha** quyidagilarga ko‘ra tasniflanadi:
 - a) tadqiqot davomida bevosita so‘rov varaqalari yordamida, telefon so‘zlashuvlari va shaxsiy suhbat yo‘li bilan olib borilishi mumkin;
 - b) davriy va maxsus adabiyotlarni o‘rganish orqali;
 - v) ma’lumotlarni telefaks yoki tayyorlangan magnitli tashuvchilar vositasida uzatish.

Bunday axborot u yoki bu muammo yoki muhitga muvofiqligi, shuningdek to‘laqlonligi va ishonarliligi bilan ajralib turadi. Ma’lumot va xabarlarni optik disklar (kompakt disklar)da va flesh disklarda uzatish so‘nggi yillarda keng ommalashdi. Ularda nafaqat matn, balki istalgan boshqa video va audio axborot yozuv ham olib boriladi.

2. **Qayta ishlash usuliga ko‘ra** ma’lumotlar birlamchi, ikkilamchi, hosila, mantiqiy xulosa va yakunlarga bo‘linadi.

Jumladan, boshlang'ich axborot odatda voqiye'likda yuz beruvchi jarayonlarni kuzatish natijasida shakllanadi va qayta ishlanmasdan qayd etiladi. Ikkilamchi axborot o'z asosiga ko'ra birlamchi ma'lumotlarga tayanadi. Hosila axborot dastlabki, ikkilamchi yoki boshqa axborotni qayta ishlash natijasidir. Shu bilan birga, tadqiqotlarda boshlang'ich axborot sifatida rejalashtirish, hisob va tahlil vazifalarini hal etish jarayonida olingan ma'lumotlar kiritiladi. Shu munosabat bilan boshlang'ich va hosila axborotni uning yuzaga kelishi muhiti va foydalanish nuqtai nazaridan ko'rib chiqish lozim.

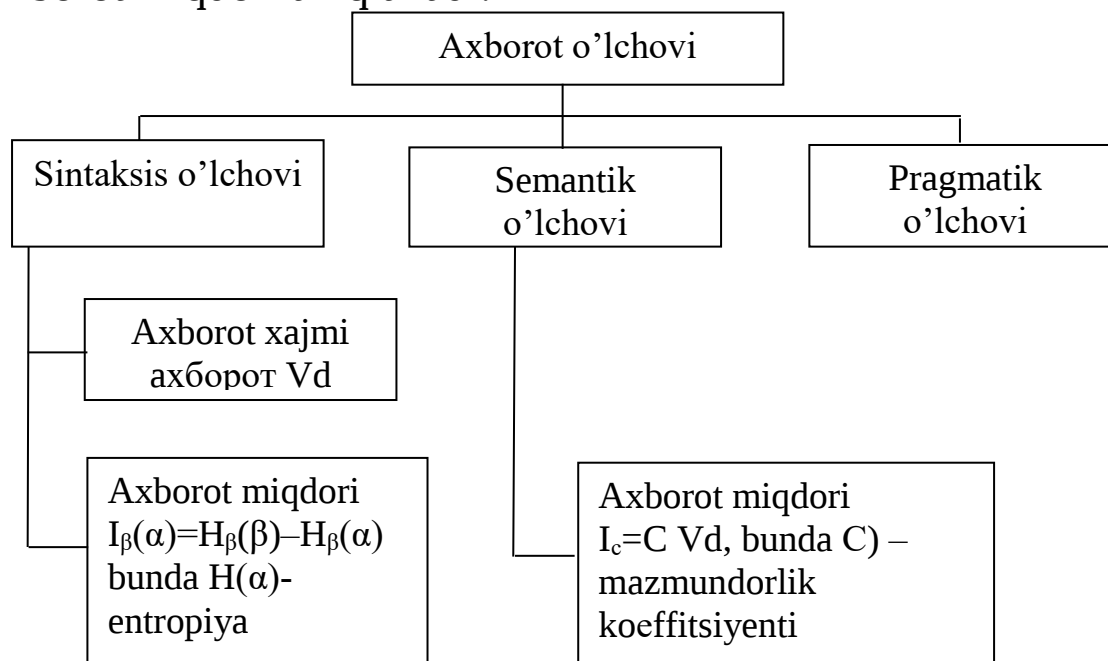


Axborotlarni o'lchash

Axborotning o'lchashda ikki parametr kiritiladi, bular axborotning miqdori (I) va xajmi (V). Ushbu parametrlar ko'riladigan yondashuvga qarab har xil tahlillanishi mumkin.

Sintaksis axborot o'lchov miqdori – bu ob'yektning ma'nosi bo'lmagan axborotlari deb hisoblanadi. Xabarlarining axborot xajmi (V_d) belgilar miqdori bilan aniqlanadi. Har xil sanoq sistemalarida elementar o'lchov birligi har xil bo'ladi. Masalan, ikkilik sanoq sistemasida bu **BIT**, o'nlik sanoq sistemasida **DIT** (masalan, 10001101 ushbu xabarning xajmi $V_d=8$ bit, 110859 ushbu xabarning xajmi $V_d=6$ dit.).

Axborot miqdorini sintaksis yondashuvda faqatgina entropiya (tizimni noaniq holati) tushunchasi orqali aniqlasa bo'ladi, chunki biror-bir tizim haqida biz qabul qilgan axborot ushbu tizim haqidagi bilimimizni o'zgartiradi. Ushbu konsepsiya quyidagicha beriladi, faraz qilamiz, α tizimi haqidagi boshlang'ich ma'lumotga egamiz. Tizimni noaniqlik holatini $H(\alpha)$ funksiyasi orqali aniqlaymiz. Qo'shimcha β ma'lumot qabul qilinganda biz qo'shimcha axborotga $I_\beta(\alpha)$ ega bo'lamiz. Natijada tizimni noaniqlik holati $H_\beta(\alpha)$ ga teng bo'ladi. Demak tizim haqidagi axborot miqdori $I_\beta(\alpha)$ quyidagicha aniqlanadi $I_\beta(\alpha)=H_\beta(\beta)-H_\beta(\alpha)$, ya'ni noaniqliqni o'zgarishi bilan axborot miqdori aniqlanadi.



Agar tizim N holatga ega bo'lsa Shennon formulasi bo'yicha uning entropiyasi teng bo'ladi:

$$H(\alpha) = - \sum_{i=1}^N P_i \log P_i$$

Bu yerda P_i - tizimning i -holatda bo'lishining ehtimoli.

Asosiy mantiqiy amallarni bajarish qoidalar

Faqat chin yoki yolg'on qiymat qabul qila oladigan darak gaplarga mulohazalar deb aytamiz.

Demak, har bir mulohaza ma'lum holatda chin yoki yolg'on qiymatga ega. Bundan keyin, chin qiymatni 1 va yolg'on qiymatni 0 bilan belgilaymiz.

$x_1, x_2, \dots, x_n$ ta o'zgaruvchi mulohaza berilgan bo'lsin. Bularning har qaysisi chin va yolg'on qiymatlarni qabul qiladi. Shuning uchun quyidagi qiymatlar satrini tuzish mumkin:

0, 0, ..., 0,
1, 0, ..., 0,
0, 1, ..., 0,
.....
1, 1, ..., 1.

Demak, o'zgaruvchilar soni n ta bo'lsa, u vaqtda $C_n^0 + C_n^1 + C_n^2 + \dots + C_n^n = 2^n$ ta qiymatlar satriga ega bo'lamiz.

$x_1, x_2 : 2^2 = 4$ ta qiymatlar satri.

$x_1, x_2, x_3 : 2^3 = 8$ ta qiymatlar satri.

Muloxazalar ustida maxsus amallar bajariladi va buning natijasida yana muloxazalar hosil bo'ladi. Bu amallarga logik (mantiqiy) amallar nomi berilgan. Bu amallar quyidagilardir.

Inkor amali. x mulohazaning inkori deb atalgan $\bar{x}$ mulohaza, x mulohaza "1" qiymatni qabul qilganda, $\bar{x}$ mulohaza "0" qiymatni qabul qiladi va aksincha.

Inkor amali oddiy tildagi manfiy sifatdosh "emas" ga to'g'ri keladi.

Inkor amalining chinlik jadvali quyidagi ko'rinishida bo'ladi:

x	$\bar{x}$
0	1
1	0

Kon'yunksiya (mantiqiy ko'paytma) amali. “ Va ” bog'lovchisiga mos keluvchi mantiqiy amalga kon'yunksiya amali deb aytamiz. x va y mulohazalarning kon'yunksiyasi $x \wedge y$ mulohaza, x va y mulohazalar chin bo'lgandagina chin qiymatni qabul qilib, qolgan hollarda esa, yolg'on qiymatni qabul qiladi.

Kon'yunksiya ta'rifini quyidagi chinlik jadvali ko'rinishida yozish mumkin:

	y	$x \wedge y$
	0	0
	1	0
	0	0
	1	1

Diz'yunksiya (mantiqiy yig'indi) amali. Rad etmaydigan ma'noda ishlatiladigan “yoki” bog'lovchisiga mos keladi. Ikki x va y mulohazaning diz'yunksiyasi $x \vee y$ murakkab mulohaza bo'lib, u faqat x va y yolg'on bo'lgandagina yolg'on qiymat qabul qilib, qolgan hollarda chin qiymatni qabul qiladi.

Diz'yunksiya amalini quyidagi chinlik jadvali orqali ham ifodalash mumkin:

x	y	$x \vee y$
0	0	0
0	1	1
1	0	1
1	1	1

Implikasiya amali. “agar....., u holda.....” bog‘lovchisiga mos keladi. Ikki x va y mulohazalarning implikasiyasi $x \rightarrow y$ deb shunday mulohazaga aytiladiki, u faqat x chin va y yolg‘on bo‘lgandagina yolg‘on bo‘lib, qolgan hamma hollarda chindir.

Implikasiya ta’rifini quyidagi chinlik jadvali ko‘rinishida bo‘ladi:

x	y	$x \rightarrow y$
0	0	1
0	1	1
1	0	0
1	1	1

Ikvivalentlik (tengkuchlilik) amali. “shunda va faqat shundagina, qachonki”, bog‘lovchisiga mos keladi. Ikvivalensiya $x \leftrightarrow y$ amali x va mulohazalar bir xil qiymat qabul qilgandagina chin, boshqa hollarda u yolg‘on qiymat qabul qiladi.

Bu ta’rifni quyidagi chinlik jadvali bilan ifodalash mumkin:

x	y	$x \leftrightarrow y$
0	0	1
0	1	0
1	0	0
1	1	1

Sheffer amali (shtrixi). Faqat x va y mulohazalar chin bo‘lgandagina, $x|y$ mulohaza yolg‘on dir.

Bu ta’rifni chinlik jadvali quyidagicha bo‘ladi:

x	y	$x y$
0	0	1
0	1	1

1	0	1
1	1	0

Pirs amali (strekasi). Faqat x va y mulohazalar yolg'on bo'lgandagina, $x \downarrow y$ mulohazachindir.

Bu ta'rifni chinlik jadvali quyidagicha bo'ladi:

x	y	$x \downarrow y$
0	0	1
0	1	0
1	0	0
1	1	0

Topshiriq variantlari

Quyidagi misollarda keltirilgan formulalarning qiymati chinlik jadvali yordamida aniqlansin.

1. $\overline{\overline{x \vee y} \rightarrow \overline{x \& y}}, \overline{p_1 \rightarrow (p_1 \rightarrow p_2)}$
2. $(x \wedge z) \vee (x \wedge \overline{z}) \rightarrow (y \wedge z) \rightarrow (\overline{x} \wedge y \wedge z)$
3. $((p \rightarrow q) \& (q \rightarrow r)) \rightarrow (p \rightarrow r)$
4. $\overline{(x \rightarrow z) \rightarrow ((y \rightarrow z) \rightarrow (x \vee y \rightarrow z))}$
5. $(p_1 \rightarrow p_2) \rightarrow ((p_1 \vee p) \rightarrow (p_2 \vee p))$
6. $\overline{(p_1 \rightarrow (p_2 \rightarrow p_3)) \rightarrow ((p_1 \rightarrow p_2) \rightarrow (p_1 \rightarrow p_3))}$
7. $\overline{(p_1 \rightarrow p_2) \rightarrow ((p_1 \wedge p) \rightarrow (p_2 \wedge p))}$
8. $(x \leftrightarrow y) \& (x \vee y) \rightarrow (z \rightarrow x)$
9. $(x \rightarrow y) \& (y \rightarrow z) \rightarrow (z \rightarrow x)$
10. $\overline{(x \vee \overline{y} \rightarrow (z \rightarrow y \vee \overline{y} \vee x)) \& (x \vee x \rightarrow (x \rightarrow x))} \rightarrow y$
11. $\overline{(x \& x \& \overline{x} \rightarrow y \& \overline{y} \rightarrow z) \vee x \vee (y \& z) \vee (y \& \overline{z})}$
12. $\overline{(x \& (y \vee z \rightarrow y \vee z)) \vee (y \& x \& \overline{y}) \vee x \vee (y \& x \& \overline{x})}$

13. $(x \rightarrow y) \& (y \rightarrow z) \rightarrow (x \rightarrow z)$
14. $(x \wedge z) \vee (x \wedge \bar{z}) \vee (y \wedge z) \vee (\bar{x} \wedge y \wedge z)$
15. $(x \wedge z) \vee (x \wedge \bar{z}) \vee (y \wedge z) \vee (\bar{x} \wedge y \wedge z)$
16. $((p \wedge q) \leftrightarrow q) \leftrightarrow (q \rightarrow p)$
17. $((p \rightarrow q) \& (q \rightarrow r)) \rightarrow (p \rightarrow r)$
18. $\overline{(x \rightarrow z) \rightarrow ((y \rightarrow z) \rightarrow (x \vee y \rightarrow z))}$
19. $\overline{(p_1 \rightarrow p_2) \rightarrow ((p_1 \vee p) \rightarrow (p_2 \vee p))}$
20. $\overline{(p_1 \rightarrow p_2) \rightarrow ((p_1 \wedge p) \rightarrow (p_2 \wedge p))}$
21. $((p \rightarrow q) \& (q \rightarrow r)) \rightarrow (p \rightarrow r)$
22. $(x \rightarrow y) \& (y \rightarrow z) \rightarrow (z \rightarrow x)$
23. $\overline{(x \vee \bar{y} \rightarrow (z \rightarrow y \vee \bar{y} \vee x)) \& (x \vee x \rightarrow (x \rightarrow x))} \rightarrow y$
24. $\overline{(x \& x \& \bar{x} \rightarrow y \& \bar{y} \rightarrow z) \vee x \vee (y \& z) \vee (y \& z)}$
25. $\overline{(x \& (y \vee z \rightarrow y \vee z)) \vee (y \& x \& \bar{y}) \vee x \vee (y \& x \& \bar{x})}$
26. $(x \rightarrow y) \& (y \rightarrow z) \rightarrow (x \rightarrow z)$

KOMPYUTERLARNING ARIFMETIK ASOSLARI. SANOQ SISTEMALARI (2LIK, 8LIK, 16LIK), ULARDA AMALLARNI BAJARISH

Nazariy qism. Sanoq sistemasi

Har qanday hisoblash mashinalarining arifmetik asosi bo'lgan sanoq sistemalari bilan tanishamiz. Shu kungacha matematika darslarida turli-tuman hisoblarni o'nta raqamdan, ya'ni 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 raqamlaridan foydalanib bajarib keldik. Shuning uchun xam bu o'nlik sanoq sistemasi deb ataladi. Sanoq sistemasida sonlarni yozish uchun qo'llaniladigan raqamlar soni ushbu sanoq sistemasining asosi deb yuritiladi. Kundalik xayotimizda qo'llanib kelinayotgan sanoq sistemasining asosi 10 ga teng.

Umuman olganda sanoq sistemalarini ikki, pozision (o'rinli) va pozision bo'lmagan (o'rinsiz) turga ajratish mumkin. Agar biror sanoq sistemasida raqamlar qiymati tushish joyiga (pozisiyasiga) qarab belgilansa u xolda bunday sistema pozision sanoq sistemasi, aks xolda pozision bo'lmagan sanoq sistemasi deyiladi.

Masalan Rim sanoq sistemasidan boshqa sanoq sistemalarining hammasi pozision, rim sanoq sistemasi esa pozision bo'lmagan sanoq sistemasiga misol bo'ladi. Xaqiqatdan, Rim sanoq sistemasida o'ttiz besh quyidagicha yoziladi.

X	X	X	V	
				qiymati besh
				qiymati o'n
				qiymati o'n
				qiymati o'n

Bu yerda foydalangan X raqami uchta bo'lishiga qaramasdan, hammasi xam o'n qiymatga teng, ya'ni raqamning qiymati uning turish o'rniga bog'liq emas.

O'nlik sanoq sistemasida yozilgan 222 sonini olaylik:

2	2	2	
			ikki birlik
			ikki o'nlik
			ikki yuzlik

Bu sonda keltirilgan uchta ikkining qiymatlari turlichadir, ya'ni

o'ngdan birinchisi ikki birlikni, ikkinchisi ikki o'nlikni, uchinchisi ikki yuzlikni ifodalaydi. ularni qiymatlari turish o'rniga qarab belgilanayapti. Shuning uchun u o'nlik sanoq sistemasi pozision sistemadir. Xuddi shunday ikkilik, sakkizlik, o'n oltilik va boshqa sanoq sistemalari ham pozision sistemagan misol bo'la oladi.

Asosi undan katta bo'lgan sanoq sistemalarida qaysi raqamlar ishlatiladi degan savol tug'iladi. Masalan, o'nlilik sanoq sistemasida biz bilgan o'nta raqam yetarli emas, shuning uchun yana 6 raqami kerak bo'ladi. bular xam o'n oltilik sanoq sistemasida bitta raqamlar uchun A B C D T F belgilarni kiritsak, o'n oltita raqamga ega bo'lamiz: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A,B C D T F. "16" soni esa 10 ko'rinishda yoziladi.

Sizga ma'lumki, o'nlik sistemaning asosi bir-u nol (10) ko'rinishda yozilgan edi. O'n oltilik sanoq sistemasining asosini xar bir-u nol (10) ko'rinishda yozdik.

Bundan keyin turli sanoq sistemalarida yozilgan sonlarni bir-biridan farq qilish uchun mos sonlarning indeksida sistema asosini ko'rsatamiz. Masalan, 27;2,37;79A;11 yozilgan bo'lsa, bu sonlar mos ravishda o'nlik, sakkizlik, o'n oltilik, va ikkilik sanoq sistemasining sonlarini ifodalaydi.

Xar qanday asosli sanoq sistemasida qisqa yozuvda berilgan sonlarni asos darajalari bo'yicha yoyib yozish mumkin. Masalan, 451 soni $4 \times 10^2 + 5 \times 10^1 + 1 \times 10$ kabi yozish mumkin. Shu kabi quyidagilar xam o'rinli:

Shunday qilib, soni biror asosli pozision sanoq sistemasida ifodalash uchun bu sonning usha sistema asosining darajalari bo'yicha yoyilmasining yigindisi shaklida yozish yetarlidir. Demak, biror sanoq sistemasining asos darajalari bo'yicha yoyilmasini mos darajalarga kutarib, so'ng qo'shib chiqilsa, xosil bo'lgan son o'nlik sanoq sistemasiga utib qolar ekan.

Sonlarni bir sanoq sistemasidan boshqasiga o'tkazish. Ma'lumki kompyuterda hisoblash ishlari ikkilik sanoq sistemasida bajariladi va zarur bo'lsa, natija o'nlik sanoq sistemasida olinishi mumkin.

Hozirgi zamon kompyuterlarida o'n oltilik sanoq sistemasi keng qo'llaniladi. Endi bir sanoq sistemasidan boshqasiga o'tish bilan tanishaylik, ya'ni ixtiyoriy asosli, masalan r asosli sanoq

sistemasidan ixtiyoriy k asosli sanoq sistemasiga o'tishni qanday amalga oshirish kerak, degan savolga javob beramiz.

Odatda, r asosli sanoq sistemasidan k asosli sanoq sistemasiga o'tish uchun oraliqda o'nlik sanoq sistemasidan foydalaniladi. Bunga sabab biz doim o'nlik sanoq sistemasida ishlab keldik va unga ko'nikib ketganmiz. Shuning uchun dastlabki r asosli sanoq sistemasidan o'n asosli sanoq sistemasiga, so'ngra esa o'n asosli sanoq sistemasidan k asosli sanoq sistemasiga o'tiladi.

Avval berilgan r asosli sanoq sistemasidan o'n asosli sanoq sistemasiga o'tishga doir misollar keltiramiz.

1-misol. 101 sonini o'nlik sanoq sistemasiga o'tkazing;

$$101_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 1 \cdot 4 + 0 \cdot 2 + 1 \cdot 1 = 4 + 1 = 5$$

Demak, $101_2 = 5_{10}$

2-misol. A9 sonini o'nlik sanoq sistemasiga o'tkazing;

$$A9_{16} = A \cdot 16^1 + 9 \cdot 16^0 = 10 \cdot 16 + 9 \cdot 1 = 160 + 9 = 169_{10}$$

Demak, $A9_{16} = 169_{10}$

Endi berilgan o'nlik sanoq sistemasidagi butun sonni boshqa biror asosli sanoq sistemasiga o'tkazishni ko'ramiz. Buning uchun ketma-ket bo'lish usulidan foydalaniladi.

O'nlik sanoq sistemasidagi aralash sonlarning kasr qismini xam boshqa asosli sanoq sistemasiga o'tkazish mumkin. Buning uchun sonning kasr qismi sanoq sistemasining asosiga ketma-ket ko'paytiriladi.

Yangi sanoq sistemasidagi kasr ko'paytirish natijasida xosil bo'lgan butun qismdagi raqamlar ketma-ketligi bilan ifodalanadi. Ketma-ket ko'paytirish jarayoni kasr qismi nolga teng bulguncha davom ettiriladi.

Ko'rib utilgan misollarda kasr qism nollarga aylangan. Lekin xar qanday sonlarda ham osongina nollar chiqavermaydi. Xosil bo'layotgan kasr son cheksiz davriy kasrlardan iborat bo'lib qolishi ham mumkin. Bunday hollarda ko'paytirish birinchi holda bir davr hosil bo'lguncha, ikkinchi holda zarur bo'lgan aniqlikka erishguncha davom ettiriladi.

Shunday qilib, o'nlik sanoq sistemasida berilgan aralash sonlarni biror asosli sanoq sistemasiga o'tkazish uchun dastlab ketma-ket bo'lish bilan butun qismini, so'ngra ketma-ket ko'paytirish bilan

kasr qismini o'tkazib, javobni har ikkalasidan foydalanib yozilar ekan.

Umuman r sonli sanoq sistemasidan k asosli sanoq sistemasiga o'tish uchun avval r asosli sanoq sistemasidan o'n asosli sanoq sistemasiga, so'ngra undan k asosli sanoq sistemasiga o'tish qoidasidan foydalanilar ekan, ya'ni bunda birinchi bosqichda asos darajalari bo'yicha yoyib chiqish, keyingisida ketma-ket bo'lish (butun sonda) yoki ketma-ket ko'paytirish (kasr sonda) usullaridan foydalaniladi.

O'n oltilik sanoq sistemasidan ikkilik sanoq sistemasiga va aksincha oson o'tish mumkin. Buning uchun **tetrada** (to'rtta ikkilik sanoq sistemasidagi raqam) lardan foydalanilsa bo'ladi. To'rtta nol yoki bir raqam yordamida 0 dan 15 gacha sonni yozish mumkin. Shuning uchun xam ikkilik va o'n oltilik sanoq sistemi orasidagi bog'lanishni amalga oshirish mumkin. Haqiqatan, to'rtta raqamdan iborat ikkilik sanoq sistemasidagi eng katta son

$1111_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0 = 8 + 4 + 2 + 1 = F_{16} = 15_{10}$ kabi bo'ladi.

Yana bir masalani qaraylik tetrada yordamida 101010,1101 sonini o'n oltilik sanoq sistemasiga o'tkazing. Buning uchun quyidagi jadvaldan foydalanamiz.

SANOQ SISTEMALARI							
2	3	4	5	6	8	10	16
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1
10	2	2	2	2	2	2	2
11	10	3	3	3	3	3	3
100	11	10	4	4	4	4	4
101	12	11	10	5	5	5	5
110	20	12	11	10	6	6	6
111	21	13	12	11	7	7	7
1000	22	20	13	12	10	8	8
1001	100	21	14	13	11	9	9
1010	101	22	20	14	12	10	A
1011	102	23	21	15	13	11	B
1100	110	30	22	20	14	12	C
1101	111	31	23	21	15	13	D
1110	112	32	24	22	16	14	E
1111	120	33	30	23	17	15	F
10000	121	100	31	24	20	16	10

Masalani yechish: Buning uchun verguldan boshlab butun qismini chapga qarab, kasr qismini unga qarab to'rttadan xonalarga ajratib chiqamiz, yetmagan qismini nollar bilan to'ldiramiz va so'ngra to'rtliklarga mos o'n oltilik raqamlarni qo'yib chiqamiz, ya'ni

$$\begin{array}{ccc} 0010 & 1010 & 1101 \\ \text{----} & \text{----}, & \text{----} \\ 3 & A & D \end{array} = 3AD$$

Topshiriq variantlari

1. 10 lik sanoq sistemasidan 2, 8, 16 lik sanoq sistemasiga o'tkazing.

I	II	III	IV
1) 107,85	1) 139,55	1) 128,24	1) 113,27
2) 1,125	2) 203,15	2) 102,55	2) 239,05
3) 10,75	3) 73,85	3) 37,95	3) 69,48
4) 39,95	4) 158,75	4) 264,85	4) 163,58
5) 87,17	5) 351,125	5) 125,86	5) 401,245
6) 185,93	6) 132,65	6) 98,42	6) 186,25
7) 258,18	7) 97,365	7) 108,65	7) 144,625

2. 2, 8, 16 lik sanoq sistemasidan 10 lik sanoq sistemasiga o'tkazing.

I	II	III	IV
1) 110111,11 ₂	1) 10101101,110 ₂	1) 101111101,101 ₂	1) 1001101,111 ₂
2) 1110110,101 ₂	2) 1100010,111 ₂	2) 10001010,1101 ₂	2) 11010,1011 ₂
3) 104,17 ₈	3) 2454,26 ₈	3) 574,23 ₈	3) 6602,32 ₈
4) 1625,24 ₈	4) 514,14 ₈	4) 1023,265 ₈	4) 123,25 ₈
5) 2C3,C4 ₁₆	5) 3D1,CF ₁₆	5) AFD,E8 ₁₆	5) DAF,B10 ₁₆
6) 7AD,B4 ₁₆	6) 2AB,4A ₁₆	6) 5E3,AB ₁₆	6) 4D2,BA ₁₆

3. 2 lik sanoq sistemasidan 8, 16 lik sanoq sistemasiga o'tkazing.

I	II	III	IV
1) 1011100,1011 ₂	1) 1010110,101 ₂	1) 1110101,1001 ₂	1) 10101,10111 ₂
2) 110110,11 ₂	2) 110001,111 ₂	2) 1001010,11001 ₂	2) 1101110,11101 ₂
3) 1625,24 ₈	3) 514,14 ₈	3) 1023,265 ₈	3) 123,25 ₈
4) 7AD,B4 ₈	4) 2AB,4A ₁₆	4) 5E3,AB ₁₆	4) 4D2,BA ₁₆
5) 379,29 ₁₀	5) 86,254 ₁₀	5) 207,255 ₁₀	5) 296,36 ₁₀

4. 8, 16 lik sanoq sistemasidan 2 lik sanoq sistemasiga o'tkazing.

I	II	II	IV
1) 574,23 ₈	1) 6602,32 ₈	1) 2454,26 ₈	1) 104,17 ₈
2) 1023,265 ₈	2) 123,25 ₈	2) 514,14 ₈	2) 1625,24 ₈
3) AFD,E8 ₁₆	3) DAF,B10 ₁₆	3) 3D1,CF ₁₆	3) 2C3,C4 ₁₆
4) 5E3,AB ₁₆	4) 4D2,BA ₁₆	4) 2AB,4A ₁₆	4) 7AD,B4 ₁₆
5) 110111,11 ₂	5) 1010110,110 ₂	5) 1011111,101 ₂	5) 1001101,111 ₂
6) 1110110,101 ₂	6) 1100010,111 ₂	6) 00001010,1101 ₂	6) 11010,1011 ₂

ALGORITMLAR, XOSSALARI, ULARNI IFODALASH USULLARI

Nazariy qism. Algoritm tushunchasi.

Algoritm — ijrochi uchun ma'lum bir masalani yechishga qaratilgan ko'rsatmalarning aniq ketma-ketligi.

Algoritm — informatika va matematikaning asosiy tushunchalaridan hisoblanadi.

Algoritm so'zi va tushunchasi *algorithmi* so'zidan olingan bo'lib, u IX (783 yilda to'g'ilgan) asrda yashab ijod etgan buyuk bobokolonimiz Muxammad al-Xorazmiy nomi bilan uzviy bog'liq bo'lib, uning arifmetikaga bag'ishlangan «Al jabr va al muqabala» nomli asarining dastlabki saxifasidagi «Dixit Algoritmici» («Dediki Ad-Xorazmiy» ning lotincha ifodasi) degan so'zlardan kelib chiqqan.

Al-Xorazmiy birinchi bo'lib o'nlik sanoq sistemasining prinsiplarini va unda turli amallar bajarish qoidalarini asoslab berdi. Bu esa hisoblash ishlarini ixchamlashtirish va osonlashtirish imkonini yaratdi. Chunki bu bilan usha davrda qo'llanib kelingan Rim raqamlari va sonlarni so'z orqali yozib bajarishdagi noqulayliklar bartaraf etildi.

Algoritm nima? Dastlab algoritm deyilganda o'nlik sanoq sistemasidagi sonlar ustida turli arifmetik amallar bajarish qoidalari tushinib kelingan. Umuman olganda, uni aniq ta'riflash mushkul. Algoritm deganda biror maqsadga erishishga yoki qandaydir masalani yechishga qaratilgan buyruqlarning aniq, tushinarli, chekli xamda to'lik tizimi, aniq natijaga olib keladigan amallarning cheklangan ketma-ketligi tushiniladi.

Algoritimning xizmati nimadan iborat? Aytaylik, kimdir qandaydir masalani yechishni uylab topib va uni boshqalarga aytmoqchi bo'lsa, u xolda u uylab topgan yechimni shunday tasvirlashi kerakki, natijada boshqalar xam masalani to'g'ri yechishsin. Shuning uchun tasvir bir necha talablarga buysunishi kerak. Agar yechimning tasviri aniq bo'lmasa, ya'ni mujmal bo'lsa, u xolda shu tasvirga asosan boshqa javobni olish mumkin. Chunki

xar kim masala yechimning tasvirini noaniq joyini uzicha aniqlashtirishi mumkin. Bunday tasvirni algoritim deb bo'lmaydi.

Xar kuni necha martalab bajaradigan ishimiz algoritimga misol bo'la oladi. Algoritmni ishlab chiqish uchun avvalo masalaning yechish yo'lini yaxshi tasavvur qilib olish, keyin esa uni formallashtirish, ya'ni aniq qoidalar ketma-ketligi ko'rinishida yozish kerak.

Algoritmning xossalari.

Algoritmlarning asosiy xossalari quyidagilardan iborat:

① **Tushunarlilik.** Algoritm ijrochisi buyruqlar ketma-ketligini qanday bajarishni aniq bilishi kerak. Algoritmning ijrochisi xamma vaqt xam inson bo'lavermaydi. Ijrochiga tavsiya etilayotgan ko'rsatmalar uning uchun tushinarli bo'lishi kerak, aks xolda ijrochi xar qanday amalni xam bajara olmaydi.

② **Diskretlik.** Algoritm ijrochisi masalani yechish jarayonini aloxida va sodda qadamlar ketma-ketligini bajarish deb tushunishi kerak. Bu xossaning mazmuni – algoritmlarni doimo chekli qadamlardan iborat qilib bulaklash imkoniyati mavjudligidadir. Boshqacha aytganda, uni chekli sondagi oddiy ko'rsatmalar ketma-ketligi shaklida ifodalash mumkin. Algoritmning bu xossasi yuqorida keltirilgan misollarda yaqqol ko'rinib turibdi. Agar kuzatilayotgan jarayonlarni chekli qadamlardan iborat qilib bulaklay olmasak, u xolda uni algoritim deb bo'lmaydi.

③ **Aniqlik.** Algoritmning xar biri qoidasi, undagi amallar va buyruqlar bir ma'noli bo'lishi kerak. Shu xossaga asosan algoritim ijrochisi buyruqlar ketma-ketligini mexanik bajarish imkoniyatiga ega bo'ladi. Ijrochiga berilayotgan kursatmalar aniq mazmunda bulishi kerak. Chunki, kursatmadagi noaniqliklar muljaldagi maqsadga erishishga olib kelmaydi.

④ **Natijaviylik.** Bu xossaning mazmuni shundan iboratki, xar qanday algoritmning ijrosi oxir-oqibat ma'lum bir yechimga kelishi kerak. Xar bir algoritim chekli sondagi qadamlardan keyin, albatta, natija berishi shart. Bajariladigan amallar ko'p bo'lsa xam, baribir natijaga olib kelishi kerak. Chekli qadamdan keyin quyilgan masala yechimga ega emasligini aniqlash xam natija hisoblanadi. Agar ko'rilyotgan jarayon cheksiz davom etib, natija bermasa, uni

algoritm deb ayta olmaymiz.

❶ **Ommaviylik.** Masalani yechish algoritmi umumiy xollar uchun yaratiladi, ya'ni faqatgina boshlang'ich qiymatlari bilan farqlanuvchi bir turdagi masalalar sinfi uchun tuziladi. Bunda boshlang'ich qiymatlar algoritmning qiymatlar qabul qilishi mumkin bo'lgan sohadan olinadi. Xar bir algoritm mazmuniga ko'ra bir turdagi masalalarning barchasi uchun xam o'rinli bulishi kerak, ya'ni masalaning boshlang'ich ma'lumotlari qanday bo'lishidan qat'iy nazar algoritm shu xildagi xar qanday masalani yechishga yaroqlidir.

Algoritmni tasvirlash usullari

Amaliyotda algoritmlarni tasvirlashning keng tarqalgan usullari quyidagilar:

1. **So'zlar yordamida** (og'zaki nutqda ishlatiladigan so'zlar) Algoritmning so'z orqali berilishi. Bunda ijrochi uchun beriladigan xar bir ko'rsatma so'zlar orqali buyruq mazmunida beriladi;

2. **Algoritmlarning formulalar yordamida berilishi.** Algoritmning formulalar bilan berilish usulidan matematika, fizika, Kimyo va boshqa aniq fanlarni o'rganishda ko'proq foydalaniladi. Masalan, uchburchakning yuzini uning asosi va balandligi bo'yicha hisoblash formulasi: $S=(h*a)/2$

3. **Algoritmlarning jadval ko'rinishida berilishi.** Algoritmni bu ko'rinishida tasvirlanishidan xam ko'p foydalaniladi.

4. **Grafik usulda** (grafik belgilar yordamida) Algoritmning grafik (blok-sxema) shaklida tasvirlanishi. Algoritmning blok-sxema ko'rinishdagi tasvirida geometrik figuralar shaklidagi oddiy elementlardan foydalaniladi;

5. **Dastur ko'rinishida** (dasturlash tillariga oid xizmatchi so'zlar, operator va funksiyalar yordamida). Algoritmlarning dastur shaklida ifodalanishi. Millionlab kompyuterlarning keng tarqalib ketishi algoritmlarning dastur tarzida tasvirning keng ommalashib ketishiga katta turtki berdi. Sababi shundaki, kompyuterlar doimo dastur orqali boshqariladi.

Algoritmlarni so'zlar yordamida tasvirlash.

Algoritmlarni so'zlar yordamida tasvirlashda bajariladigan buyruqlar va ko'rsatmalar ketma-ket og'zaki nutqda

ishlatiladigan soʻzlar orqali yoziladi.

Masalan, ikki sonning **eng katta umumiy boʻluvchisini (EKUB) topish** algoritmi quyidagicha yozilishi mumkin:

1q. Ikkita sonni kiriting;

2q. Agarda bu sonlar teng boʻlsa, u xolda ulardan birini javob sifatida oling va ishni toʻxtating, aks xolda esa davom ettiring;

3q. Ikkita son ichida kattasini aniqlang;

4. Katta va kichik sonlarning ayirmasini katta son bilan almashtiring;

5q. Algoritmni 2-qadamdan boshlab qaytaring.

Keltirilgan algoritmni xar qanday natural sonlarning EKUBini topish uchun ishlatish mumkin.

Algoritmnlarni soʻzlar yordamida tasvirlashning bir qancha kamchiligi mavjud boʻlib, aksariyat xollarda algoritmnlarni tasvirlashda bu usuldan foydalanilmaydi.

Algoritmnlarni grafik usulda tasvirlash.

Algoritmnlarni grafik usulda tasvirlashda xar bir amal bir yoki bir nechta xarakatni ifodalovchi oʻzaro bogʻlik funksional bloklar ketma-ketligi orqali tasvirlanadi.

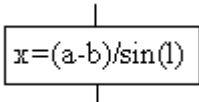
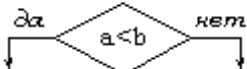
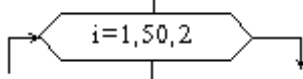
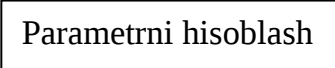
Algoritmning bunday tasvirlash usuli algoritm sxemasi yoki **blok-sxema** deb ataladi.

Blok-sxemada xar bir xarakter toʻrini (boshlangich qiymatlarni kiritish, ifodalar qiymatlarini hisoblash, shartlarni tekshirish, amallarni takrorlashni boshqarish, qayta ishlashni tugatish va x.k.) maʼlum bir geometrik figura orqali ifodalanadi.

Blokli belgilar (geometrik figuralar) chiziqlar orqali bogʻlanadi (bunda qaysi amal oldin, qaysinisi keyin bajarilishi koʻrsatiladi). Nisbatan murakkab masalalarni yechishda algoritm dan muayan EHM tilidagi dasturga oʻtish juda qiyin. Bunday bevosita oʻtishda algoritmning aloxida qismlari orasidagi bogʻlanish yuqoladi, algoritm tarkibining asosiy va muxim boʻlmagan qismlarini farqlash qiyin boʻlib qoladi. Bunday sharoitda keyinchalik aniqlash va tugʻrilash ancha vaqt talab qiladigan xatolarga osongina yoʻl quyish mumkin. Odatda, algoritm bir necha marta ishlab chiqiladi, baʼzan

xatolarni tugʻrilash, algoritm tarkibini aniqlashtirish va tekshirish uchun bir necha marta orqaga qaytishga toʻgʻri keladi. Algoritm ishlab chiqishning birinchi bosqichida algoritmni yozishning eng qulay usuli algoritmni blok-sxema koʻrinishda ifodalashdir.

Jadvalda blok-sxemada ishlatiladigan bloklar aks ettirilgan.

Amallarni belgilanishi	Izox
	Oddiy xarakat
	Shart tekshirish
	Sikl (takrorlanish) boshi
	Yordamchi algoritmgaga murojaat
	Maʼlumotlarni kiritish va chiqarishning umumiy koʻrinishi
	Algoritmning boshi va oxiri
	Natijani bosmaga chiqarish

“Oddiy xarakat” belgisi orqali formulalar, xisob-kitob, oʻzlashtirish amallari ifodalaniladi. Bir nechta amallarni aloxida yoki bitta belgi orqali ifodalash mumkin.

"Shart tekshirish" bloki orqali amallar bajarilish yoʻnalishi shart bajarilishi asosida koʻrsatiladi. Bunday blokning xar birida savol, shart yoki munosabat kursatiladi.

"Sikl" bloki amallarni takrorlash uchun ishlatiladi. Blok ichida siklning boshi va oxirini kursatuvchi parametr (**i**), parametrning oʻzgarish qadami koʻrsatiladi.

"Yordamchi algoritmgaga murojaat" bloki aloxida va mustaqil ishlovchi qism dastur va yordamchi algoritmlarga murojaatni bildiradi.

DASTURLASHNING BIRINCHI QADAMLARI

Qo'llanmada dasturlarni yozish uchun asosiy til qilib C++ dasturlash tili tanlangan. Mazkur til Oliy ta'lim muassasalarida keng tarqalgan va ushbu dasturlash tili dasturlashni yuqori darajada o'rganishga mo'ljallab yaratilgan. Ushbu dasturlash tilida dasturlash usullarini yaxshilab o'rganib olib, boshqa algoritmik tillar, masalan, C#, Java tillarining sintaksis va boshqa xususiyatlarini o'rganish mumkin. Asosiysi dasturlash tilini emas balki, dasturlash tajribasini mukammal o'rganishdir. Dasturlashni zo'rlab o'rgatib yoki o'rganib bo'lmaydi. Dasturlashni bilish bu – ilm emas, balki bu san'at darajasidagi mukammal qobiliyat egasining iqtidor maxsulidir.

Dastur va dasturchilar haqida

Biz ishlaydigan har qanday dastur biron-bir shaxs yoki shaxslar tomonidan yozilgan bo'lib, ular **dasturchilar** deb ataladi. Dasturchilar dasturlarni maxsus tillarda yozishadi, ularni biz **dasturlash tillari** deb ataymiz. Bu tillar yordamida dasturchilar kompyuterga bajarish kerak bo'lgan amallarni maxsus kodlar yordamida tushuntirishadi. Bu tushuntirishlar, kompyuter nimani, qachon, qachongacha, qancha marta amalga oshirishi kerakligini aniq yo'riqnoma asosida bajarish kerakligini bildiradi. Bu yo'riqnomaning biz algoritmi deb ataymiz. Algoritm oldin biron-bir g'oya, keyinchalik qoralama yoki inson uchun tushunarli tilda qog'ozga bitilgan tilda yoki shakl (grafik ko'rinish)da bo'ladi. Shu o'rinda odamlar ham ayrim vaqtlarda qog'ozga tushurilgan algoritimga asoslanib ish qilishi haqida mavjud misollardan keltiraylik.

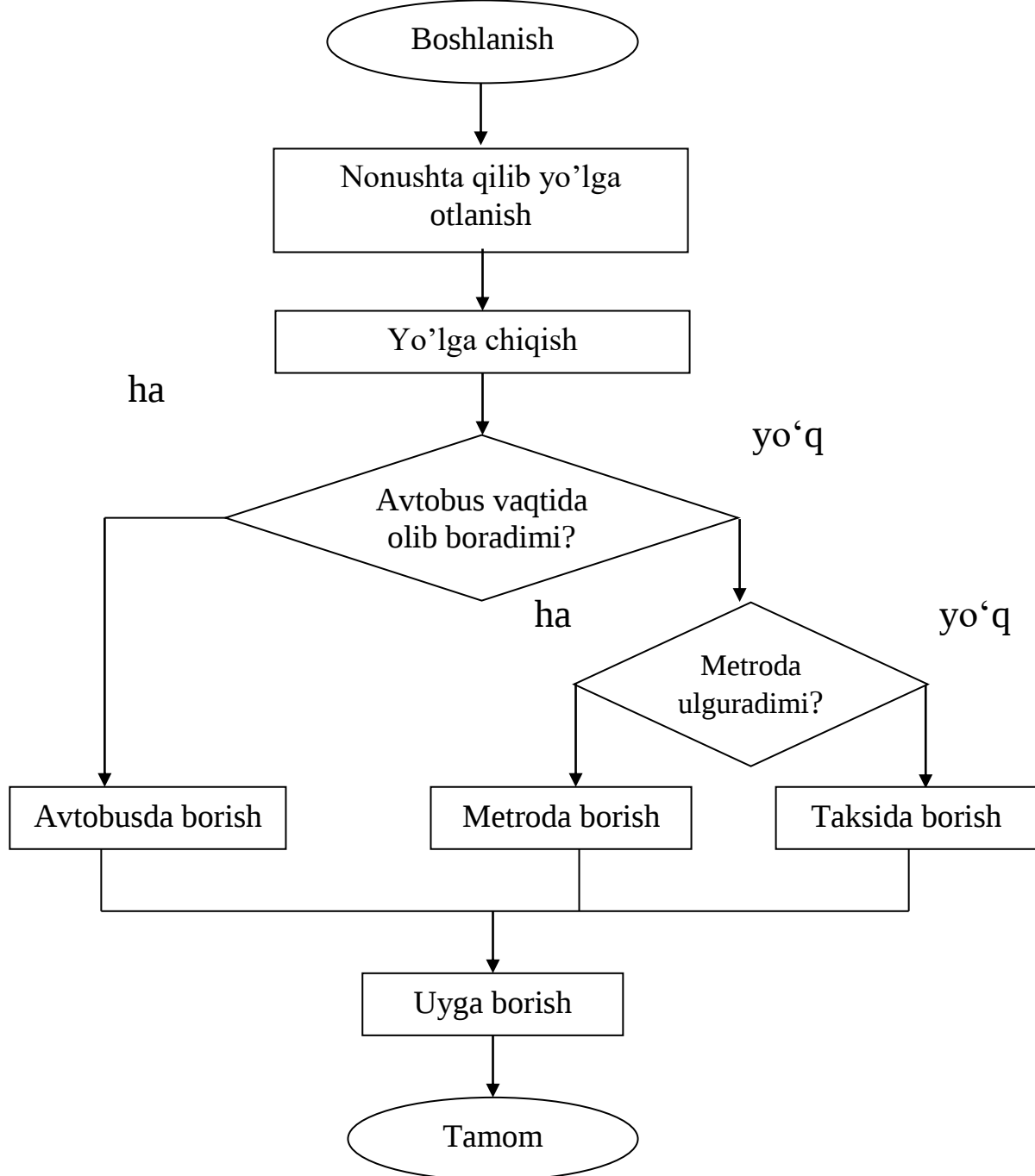
Ota-ona o'z o'g'liga dars mashg'ulotiga yetib borishi uchun zarur bir yo'riqlarni berishi mumkin.

Nonushta qilib yo'lga o'tlanish.

Yo'lga chiqish

Yo'lga chiqib avtobusga minish, agar avtobus mashg'ulotga borishni kechiktiradigan bo'lsa metroda borish, agar u ham kechgacha qolishni olib kelsa, taksida borish va mashg'ulotlar tugaganidan so'ng to'g'ri uyga qaytish.

Ko‘rib turganimizdek, bu yo‘riqni osongina usulda oddiy so‘z bilan ifodalash mumkin. Shuningdek usbu yo‘riqni blok-sxema ko‘rinishida ham ifodalashimiz mumkin.



Tavsiyalar blok-sxema ko‘rinishida berilishi

Ko‘rib turganingizdek, aynan bir masala turli xil ko‘rinishda ifodalanishi mumkin ekan.

C++ DASTURLASH TILINING TARIXI

C++ dasturlash tili C tiliga asoslangan. C esa o'z navbatida B va BCPL tillaridan kelib chiqqan. BCPL 1967 yilda Martin Richards tomonidan tuzilgan va operatsion tizimlarni yozish uchun mo'ljallangan edi. Ken Thompson o'zining B tilida BCPL ning ko'p hossalarni kiritgan va B da UNIX operatsion tizimining birinchi versiyalarini yozgan. BCPL ham, B ham tipsiz til bo'lgan. Yani o'zgaruvchilarning ma'lum bir tipi bo'lmagan - har bir o'zgaruvchi kompyuter hotirasida faqat bir bayt joy egallagan. O'zgaruvchini qanday sifatda ishlatish esa, yani butun sonmi, kasrli sonmi dasturchining vazifasi bo'lgan. C tilini Dennis Ritchie B dan keltirib chiqardi va uni 1972 yili ilk bor Bell Laboratoriyasida, DEC PDP-11 kompyuterida qo'lladi. C o'zidan oldingi B va BCPL tillarining juda ko'p muhim tomonlarini o'z ichiga olish bilan bir qatorda o'zgaruvchilarni tiplashtirdi va bir qator boshqa yangiliklarni kiritdi.

Boshlanishda C asosan UNIX tizimlarida keng tarqaldi. Hozirda operatsion tizimlarning asosiy qismi C/C++ da yozilmoqda. C mashina arxitekturasiga bog'langan tildir. Lekin yaxshi rejalashtirish orqali dasturlarni turli kompyuter platformalarida ishlaydigan qilsa bo'ladi. 1983 yilda, C tili keng tarqalganligi sababli, uni standartlash harakati boshlandi. Buning uchun Amerika Milliy Standartlar Komiteti (ANSI) qoshida X3J11 texnik komitet tuzildi. Va 1989 yilda ushbu standart qabul qilindi. Standartni dunyo bo'yicha keng tarqatish maqsadida 1990 yilda ANSI va Dunyo Standartlar Tashkiloti (ISO) hamkorlikda C ning ANSI/ISO 9899:1990 standartini qabul qilishdi. Shu sababli C da yozilgan dasturlar kam miqdordagi o'zgarishlar yoki umuman o'zgarishsiz juda ko'p kompyuter platformalarida ishlaydi. C++ 1980 yillar boshida Bjarne Stroustrup tomonidan C ga asoslangan tarzda tuzildi.

C++ tilinig tashkil etuvchilari

C++ dasturlash tilining tashkil etuvchilari ikki qismga bo'linadi: til leksemasi va til semantikasi.

Xar qanday dasturlash tillari kabi C++ dasturlash tilining xam alfavit mavjud. C++ alfavitiga quyidagi belgilar kiradi.

- Lotin alifbosining katta va kichik xarflari (A,B,...,Z,a,b,...,z)
- Raqamlar: 0,1,2,3,4,5,6,7,8,9
- Maxsus belgilar: “, {} [] () + - / % \ ; ‘ . : ? < = > _ ! & * # ~ ^
- Ko‘rinmaydigan belgilar (“umumlashgan bo‘shliq belgilari”).

Leksemalarni o‘zaro ajratish uchun ishlatiladigan belgilar (misol uchun bo‘shlik, tabulyatsiya, yangi qatorga o‘tish belgilari).

Til semantikasi leksemalardan tashkil topadi va quyidagilar kiradi:

- Identifikatorlar;
- Kalit (maksus, xizmatchi) so‘zlar;
- Literallar (o‘zgarmaslar);
- Ajratuvchilar.

Izohlarda, satrlarda va belgili o‘zgarmaslarda boshqa literalalar, masalan rus xarflari xam ishlatilishi mumkin.

C++ tilida olti xil turdagi leksemmalar ishlatiladi: erkin tanlanadigan va ishlatiladigan identifikatorlar, xizmatchi so‘zlar, o‘zgarmaslar (o‘zgarmas satrlar), amallar (amallar belgilari), ajratuvchi belgilar.

Identifikator. Identifikatorlar lotin xarflari, ostki chiziq belgisi va sonlar ketma-ketligidan iborat bo‘ladi. Identifikator lotin xarfidan yoki ostki chizish belgisidan boshlanishi lozim.

Misol uchun:

A1, _MAX, adress_01, RIM, rim

katta va kichik xarflar farqlanadi, shuning uchun oxirgi ikki identifikator bir-biridan farq qiladi.

Borland kompilyatorlaridan foydalanilganda nomning birinchi 32 xarfi, ba’zi kompilyatorlarda 8 ta xarfi inobatga olinadi. Bu holda NUMBER_OF_TEST va NUMBER_OF_ROOM identifikatorlari bir-biridan farq qilmaydi.

Xizmatchi so‘zlar. Tilda ishlatiluvchi ya’ni dasturchi tomonidan o‘zgaruvchilar nomlari sifatida ishlatish mumkin bo‘lmagan identifikatorlar **xizmatchi so‘zlar** deyiladi.

C ++ tilida quyidagi xizmatchi soʻzlar mavjud:

int	extern	else
char	register	for
float	typedef	do
double	static	while
struct	goto	switch
union	return	case
long	sizeof	default
short	break	entry
unsigned	continue	
auto	if	

Oʻzgaruvchliar (variables)

Oʻzgaruvchilarni ob'yekt sifatida qarash mumkin. C++ tilining asosiy tushunchalaridan biri nomlangan hotira qismi – ob'yekt tushunchasi hisoblanadi. Ob'yektning xususiy holi bu oʻzgaruvchidir. Oʻzgaruvchiga qiymat berilganda unga ajratilgan hotira qismiga shu qiymat kodi yoziladi. Oʻzgaruvchi qiymatiga nomi orqali murojaat qilish mumkin, hotira qismiga esa faqat uning adresi (xotira qismining tartib nomeri) orqali murojaat qilinadi. Oʻzgaruvchi nomi bu erkin kiritiladigan identifikatordir. Oʻzgaruvchi nomi sifatida xizmatchi soʻzlarni ishlatish mumkin emas.

Oʻzgaruvchilar tiplari. Oʻzgaruvchilarning quyidagi tiplari mavjuddir:

char – bitta belgi;

long char – uzun belgi;

int – butun son;

short yoki **short int** – qisqa butun son;

long yoki **long int** – uzun butun son;

float - haqiqiy son;

long float yoki **double** – ikkilangan haqiqiy son;

long double – uzun ikkilangan haqiqiy son;

Butun sonlar ta'riflanganda koʻrilgan tiplar oldiga **unsigned** (ishorasiz) ta'rifi qoʻshilishi mumkin. Bu ta'rif qoʻshilgan butun sonlar ustida amallar **mod 2n** arifmetikasiga asoslangandir. Bu yerda

n soni **int** tipi hotirada egallovchi razryadlar sonidir. Agar ishorasiz **k** soni uzunligi **int** soni razryadlar sonidan uzun bo'lsa, bu son qiymati **k mod 2n** ga teng bo'ladi. Ishorali ya'ni **signed** tipidagi sonlarning eng katta razryadi son ishorasini ko'rsatish uchun ishlatilsa **unsigned** (ishorasiz) tipdagi sonlarda bu razryad sonni tasvirlash uchun ishlatiladi.

O'zgaruvchilarni dasturning ixtiyoriy qismida ta'riflash yoki qayta ta'riflash mumkin.

Misol uchun:

```
int a, b1, ac;
```

yoki

```
int a;
```

```
int b1;
```

```
int ac;
```

O'zgaruvchilar ta'riflanganda ularning qiymatlari aniqlanmagan bo'ladi. Lekin o'zgaruvchilarni ta'riflashda initsializatsiya ya'ni boshlang'ich qiymatlarini ko'rsatish mumkin.

Misol uchun:

```
int i=0;
```

```
char c='k';
```

Typedef ta'riflovchisi yangi tiplarni kiritishga imkon beradi.

Misol uchun yangi COD tipini kiritish:

```
Typedef unsigned char COD;
```

COD simbol;

C++ da arifmetik amallar

Ko'p dasturlar ijro davomida arifmetik amallarni bajaradi. C++ dagi amallar quyidagi jadvalda berilgan. Ular ikkita operand bilan ishlatildi.

C++ dagi amal	Arifmetik operator	Algebraik ifoda	C++ dagi ifodasi:
Qo'shish	+	$h+19$	$h+19$
Ayirish	-	$f-u$	$f-u$
Ko'paytirish	*	$s*l$	$s*l$
Bo'lish	/	v/d	v/d

Modul olish	%	k mod 4	k%4
-------------	---	---------	-----

Bularning ba'zi birlarinig hususiyatlarini ko'rib chiqaylik. Butun sonli bo'lishda, yani bo'luvchi ham, bo'linuvchi ham butun son bo'lganda, javob butun son bo'ladi. Javob yahlitlanmaydi, kasr qismi tashlab yuborilib, butun qismining o'zi qoladi. Modul operatori (%) butun songa bo'lishdan kelib chiqadigan qoldiqni beradi. $x\%y$ ifodasi x ni y ga bo'lgandan keyin chiqadigan qoldiqni beradi. Demak, $7\%4$ bizga 3 javobini beradi. % operatori faqat butun sonlar bilan ishlaydi. Vergulli (float) sonlar bilan ishlash uchun "math.h" kutubhonasidagi fmod funksiyasini qo'llash kerak. C++ da qavslarning ma'nosi huddi algebradagi kabidir. Undan tashqari boshqa algebraik ifodalarning ketma-ketligi ham odatdagidek bo'ladi. Oldin ko'paytirish, bo'lish va modul olish operatorlari bajariladi. Agar bir necha operator ketma-ket kelsa, ular chapdan o'nga qarab ishlanadi. Bu operatorlardan keyin esa qo'shish va ayirish bajariladi.

Misol keltiraylik.

$$k = m * 5 + 7 \% n / (9 + x);$$

Birinchi bo'lib $m * 5$ hisoblanadi. Keyin $7 \% n$ ning qiymati topiladi va qoldiq $(9 + x)$ ifodaning qiymatiga bo'linadi. Chiqqan natija esa $m * 5$ ning natijasiga qo'shiladi. Qisqasini aytsak, amallar matematikadagi kabi. Lekin biz o'qishni osonlashtirish uchun va xato qilish ehtimolini kamaytirish maqsadida qavslarni ko'proq ishlatishimiz mumkin. Yuqoridagi misolimiz quyidagi ko'rinishga ega bo'ladi.

$$k = (m * 5) + ((7 \% n) / (9 + x));$$

Amallar jadvali

Arifmetik amallar	Razryadli amallar	Nisbat amallari	Mantiqiy amallar
+ qo'shish	& va	= = teng	&& va
- bo'lish	yoki	!= teng emas	yoki
* ko'paytirish	^ inkor	> katta	! inkor
/ bo'lish	<< chapga surish	>= katta yoki teng	
% modul	>> o'ngga	< kichik	

olish	surish		
- unar minus	~ inkor	<= kichik yoki teng	
+ unar plyus			
++ oshirish			
-- kamaytirish			
() – doirali qavs	= - oddiy qiymar berish	(tip) – tipni o‘zgartirish	& - adresni aniqlash
[] – kavadrat qavs	op= - murakab qiymat berish	sizeof - hajmni hisoblash	* - adres bo‘yicha qiymat aniqlash yoki joylash
, - vergul	? – shartli amal		

Arifmetik amallar. Amallar odatda unar ya’ni bitta operandga qo‘llaniladigan amallarga va binar ya’ni ikki operandga qo‘llaniladigan amallarga ajratiladi. Binar amallar additiv ya’ni + qo‘shuv va – ayirish amallariga, hamda multiplikativ ya’ni * ko‘paytirish, / bo‘lish va % modul olish amallariga ajratiladi. Additiv amallarining ustivorligi multiplikativ amallarining ustivorligidan pastroqdir. Butun sonni butun songa bo‘lganda natija butun songacha yahlitlanadi. Misol uchun $20/3=6$; $(-20)/3=-6$; $20/(-3)=-6$. Modul amali butun sonni butun songa bo‘lishdan hosil bo‘ladigan qoldiqqa tengdir. Agar modul amali musbat operandlarga qo‘llanilsa, natija ham musbat bo‘ladi, aks holda natija ishorasi kompilyatorga bog‘liqdir. Binar arifmetik amallar bajarilganda tiplarni keltirish quyidagi qoidalar asosida amalga oshiriladi:

short va **char** tiplari **int** tipiga keltiriladi;

Agar operandlar biri **long** tipiga tegishli bo‘lsa ikkinchi operand ham **long** tipiga keltiriladi va natija ham **long** tipiga tegishli bo‘ladi;

Agar operandlar biri **float** tipiga tegishli bo‘lsa ikkinchi operand xam **float** tipiga keltiriladi va natija ham **float** tipiga tegishli bo‘ladi;

Agar operandlar biri **double** tipiga tegishli bo'lsa ikkinchi operand ham **double** tipiga keltiriladi va natija ham **double** tipiga tegishli bo'ladi;

Agar operandlar biri **long double** tipiga tegishli bo'lsa ikkinchi operand ham **long double** tipiga keltiriladi va natija ham **long double** tipiga tegishli bo'ladi;

Unar amallarga ishorani o'zgartiruvchi unar minus – va unar + amallari kiradi. Bundan tashqari ++ va -- amallari ham unar amallarga kiradi. ++ unar amali qiymatni 1 ga oshirishni ko'rsatadi. Amalni prefiks ya'ni ++i ko'rinishda ishlatish oldin o'zgaruvchi qiymatini oshirib so'ngra foydalanish lozimligini, postfiks ya'ni i++ ko'rinishda ishlatish oldin o'zgaruvchi qiymatidan foydalanib so'ngra oshirish kerakligini ko'rsatadi.

Misol uchun i qiymati 2 ga teng bo'lsin, u holda 3+(++i) ifoda qiymati 6 ga, 3+i++ ifoda qiymati 5 ga teng bo'ladi. Ikkala holda ham i qiymati 3 ga teng bo'ladi. -- unar amali qiymatni 1 ga kamaytirishni ko'rsatadi. Bu amal ham prefiks va postfiks ko'rinishda ishlatilishi mumkin. Bu ikki amalni faqat o'zgaruvchilarga qo'llash mumkindir. Unar amallarning ustivorligi binar amallardan yuqoridir.

Mantiqiy amallar. C++ tilida mantiqiy tip yo'qdir. Shuning uchun mantiqiy amallarni butun sonlarga qo'llanadi. Bu amallarning natijalari quyidagicha aniqlanadi: x||y amali 1 ga teng agar x>0 yoki y>0 bo'lsa, aksincha 0 ga teng x&& y amali 1 ga teng agar x>0 va y>0 bo'lsa, aksincha 0 ga teng !x amali 1 ga teng agar x>0 bo'lsa, aksincha 0 ga teng Bu misollarda amallar ustivorligi oshib borish tartibida berilgandir. Inkori ! amali unar qolganlari binar amallardir.

Amallar ustivorligi

Rang	Amallar	Yo'nalish
1	() [] -> :: .	Chapdan o'ngga
2	! ~ + - ++ -- & * (tip) sizeof new delete tip()	O'ngdan chapga
3	. * ->*	Chapdan o'ngga
4	* / % (multiplikativ binar amallar)	Chapdan o'ngga

5	+ - (additiv binar amallar)	Chapdan o'ngga
6	<< >>	Chapdan o'ngga
7	< <= >= >	Chapdan o'ngga
8	= !=	Chapdan o'ngga
9	&	Chapdan o'ngga
10	^	Chapdan o'ngga
11		Chapdan o'ngga
12	&&	Chapdan o'ngga
13		Chapdan o'ngga
14	?: (shartli amal)	Chapdan o'ngga
15	= *= /= %= += -= &= ^= = <<= >>=	Chapdan o'ngga
16	, (vergul amali)	Chapdan o'ngga

C++ DA DASTUR TUZILISHI

C++ tilidagi oddiy dastur

Ekranga Hello, World! soʻzlarini chop etadigan klassik masalani qaraymiz:

```
//Hello.cpp – dastur faylining nomi
#include<iostream >
int main(){
    cout<<“\nHello, World!\n”;
}
```

Dastur matnining birinchi satri – bu ‘//’ belgisi bilan boshlanib, chop qilinmaydigan ‘satr oxiri’ belgilari tugaydigan belgilar orasidagi bir satrli izohdir. Bu ajratuvchi belgilar orasiga ixtiyoriy belgilarni joylashtirish mumkin. Mazkur misoldagi izohda berilgan dastur matni saqlanayotgan Hello.cpp fayl nomi koʻrsatilmoqda.

Dastur matnining ikkinchi satrida dasturdagi maʼlumotlarning kiritish va chiqarishning standart oqimlarini bogʻlaydigan preprocessor boʻyruqlari (direktivasi) keltirilgan. Koʻrsatilgan vositalar iostream.h faylida joylashgan (“i”(input): - kirish; “o”(output) – chiqarish; stream – oqim; “h”(head) – sarlavha). Jimlik qoidasi boʻyicha standart chiqarish oqimi ekranga chiqarishni bildiradi. Standart kiritish oqimi esa klaviaturadan maʼlumotlarni oʻqishni taʼminlaydi. Uchinchi satr main nomli funktsiya sarlavhasi hisoblanadi. C++ tilidagi ixtiyoriy bajariluvchi dasturlarda bu nomli funktsiya faqat va faqat bir marta uchrashi kerak. main nomidan oldin kelgan maxsus soʻz void – mazkur dasturdagi main funktsiyasi hech qanday qiymat qaytarmasligini bildiruvchi turni anglatadi. main dan keyingi aylana qavslar ixtiyoriy funktsiya sarlavhasi sintaksisiga mos kelishini talab etadi. Unda parametrlar roʻyxati joylashadi. Bizning misolda parametrlar yoʻq va roʻyxat boʻsh.

Ixtiyoriy funktsiyaning tanasi – bu figurali qavsga olingan tavsiflar, aniqlanish va operatorlar ketma-ketligidan tashkil topadi. Har bir tavsiflar, aniqlashlar yoki operatorlar “nuqta vergul” bilan tugashi kerak. main funktsiyasining tanasida aniq tavsiflar va aniqlashlar yoʻq va u bitta operatoridan iborat

```
cout<<“\nHello, World!\n”;
```

cout iostream.h faylining oqimi bo'lib, ma'lumotlarni ekranga chiqarishni ta'minlaydigan ob'yekt nomi hisoblanadi (chiqarishning standart oqimi). cout ob'yektiga ma'lumotlarni chiqarish uchun << amali ("joylashirish") yordamida o'zatiladi. Chiqarilishi kerak ma'lumotlar << amalidan o'ng tomonga joylashishi kerak. Hozirgi vaqtda bu "\nHello, World!\n" satridir (satrli o'zgarmas). C++ tilidagi satr qo'sh tirnoqqa olingan ixtiyoriy belgi ketma-ketligi kabi aniqlanadi. Ularning orasida ekranga chiqarilmaydigan boshqaruvchi belgilar ham bo'lishi mumkin. Masalan, '\n' – boshqaruvchi beligisi keyingi satr yangi satrdan chiqarilishini ta'minlaydi. Ya'ni, dastur Hello, World!ni yangi satrdan chiqaradi va kursorni keyingi satr boshiga o'tkazadi.

Hozirda shuni e'tirof qilish kerakki, C++ tilining asosiy xususiyatlaridan biri qo'shimcha (yuklama) yoki standart amallar harakatining nomlanishidir. << leksemasi faqat shu holatda qo'yish amali («joylashtirish») bo'ladiki, agar undan chap tomonda cout ob'yektining nomi joylashgan bo'lsa. Qolgan hollarda << belgilar juftligi chapga siljitish binar amalini bildiradi. Demak, bu dasturning yagona amalida chiqish oqimi cout qiymati o'rnatish amalidan foydanilgan, ya'ni o'ngdan << leksemasi joylashtirilgan.

Dasturni bajartirish uchun uning matnini .cpp kengaytmali faylda tayyorlash kerak; bu faylni kompilyatsiyaga jo'natish va kompilyatsiya vaqtida hosil bo'lgan sintaksis xatolarni to'zatis; xatosiz kompilyatsiya (.obj kengaytmali ob'yektli fayl hosil bo'ladi); kerakli kutubxona funksiyalarni ob'yektli faylga qo'shish va .exe kengaytmali bajariluvchi dastur modulini olish. Bajariluvchi faylni hosil qilish sxemasi 2-rasmda ko'rsatilgan, kompilyator qadamidan oldin priprocessor dastur matnini qayta ishlash qadami ko'rsatilgan. Bizning misolda priprocessor **#include** <iostream.h> direktivasini qayta ishlashga va dastur matniga display bilan almashtirish vositasi uchun bog'laydi (<< amalini saqlab turish uchun).

Agar dasturning dastlabki matni HELLO.CPP faylida tayyorlangan bo'lsa, unda priprocessor uning direktivasi bajarib to'liq dastur matni shakllantiriladi, kompilyator HELLO.OBJ ob'yektli faylini yaratadi, unda ko'rsatilgan nomi tanlanadi (jimlik qoidasiga asosan), komponovshik esa (Linker aloqa redaktori)

dasturni kutubxona funksiyalari bilan to'ldiradi, masalan, cout obyekti bilan ishlaganda, HELLO.EXE faylini ishga tushirganda, ekranda kutilgan Hello, World! frazasini ko'ramiz.

Yuqorida ko'rsatilgan harakatlarning bajarilish xususiyatlari C++ aniq kompilyatori va dastur ishlayotgan operasion tizimga bog'liq. Texnik tavsilotlarni aniq dastur mahsuloti hujjatlaridan o'rganish kerak.

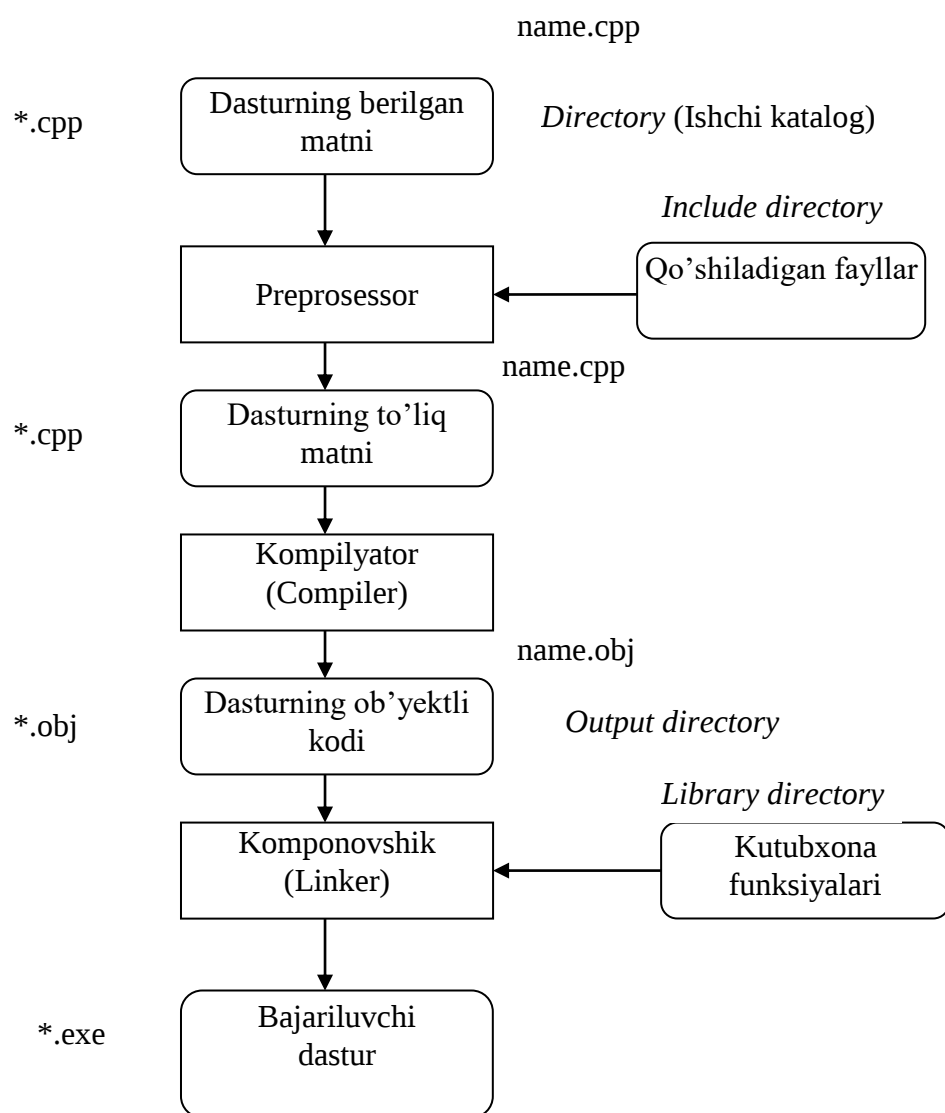
Dastur matni va preprocessor. C++ tilida matnli fayl shaklida tayyorlangan dastur uchta qayta ishlash bosqichlaridan o'tadi. Matnni preprocessor direktivalari asosida o'zgartilishi. Bu jarayon natijasi yana matnli fayl bo'lib preprocessor tomonidan bajariladi.

Kompilyatsiya. Bu jarayon natijasi mashina kodiga o'tkazilgan ob'yekтли fayl bo'lib, kompilyator tomonidan bajariladi.

Bog'lash. Bu jarayon natijasi to'la mashina kodiga o'tkazilgan bajariluvchi fayl bo'lib, bog'lagich (komponovthik) tomonidan bajariladi.

Preprocessor vazifasi dastur matnini preprocessor direktivalari asosida o'zgartirishdir.

define direktivasi dasturda bir jumlani ikkinchi jumla bilan almashtirish uchun ishlatiladi. Bu direktivadan foydalanishning sodda misollarini biz yuqorida ko'rib chiqdik.



2-rasm. Bajariluvchi dasturni tayyorlash sxemasi

include direktivasi ikki ko'rinishda ishlatilishi mumkin. **#include** "fayl nomi" direktivasi dasturning shu direktiva o'rniga qaysi matnli fayllarni qo'shish kerakligini ko'rsatadi. **#include** <fayl nomi> direktivasi dasturga kompilyator standart bibliotekalariga mos keluvchi sarlavhali fayllar matnlarini qo'shish uchun mo'ljallangandir. Bu fayllarda funksiya prototipi, tiplar, o'zgaruvchilar, o'zgarmaslar ta'riflari yozilgan bo'ladi. Funksiya prototipi funksiya qaytaruvchi tip, funksiya nomi va funksiya uzatiluvchi tiplardan iborat bo'ladi. Agar funksiya nomidan oldin **void** tipi ko'rsatilgan bo'lsa bu funksiya hech qanday qiymat qaytarmasligini ko'rsatadi. Shuni ta'kidlash lozimki bu direktiva dasturga standart biblioteka qo'shilishiga olib kelmaydi. Standart

funksiyalarning kodlari bog‘lash ya’ni aloqalarni tahrirlash bosqichida, kompilyatsiya bosqichidan so‘ng amalga oshiriladi. Kompilyatsiya bosqichida sintaksis hatolar tekshiriladi va dasturda bunday hatolar mavjud bo‘lmasa, standart funksiyalar kodlarisiz mashina kodiga utkaziladi. Sarlavhali fayllarni dasturning ixtiyoriy joyida ulash mumkin bo‘lsa ham, bu fayllar odatda dastur boshida qo‘shish lozimdir. Shuning uchun bu fayllarga sarlavhali fayl (header file) nomi berilgandir.

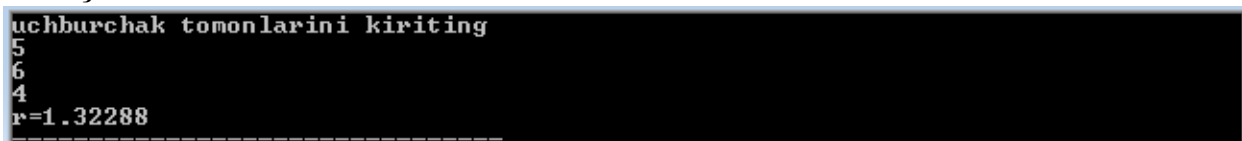
Chiziqli jarayonlarni dasturlash

Oldingi bo‘limda ko‘rib chiqilgan oddiy dasturlar operatorlarning ketma-ketligi ko‘rinishida bajariladi. Bunday algoritmlar chiziqli algoritmlar deb ataladi. Quyida chiziqli algoritmlarga misollar keltiramiz.

Misollar:

1. Tomonlar uzunligi a , b , c ga teng bo‘lgan uchburchakka ichki chizilgan aylana radiusi r ga teng. r ni hisoblash dasturini tuzing.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float a,b,c,p,r,s;
    cout<<"uchburchak tomonlarini kiriting " << endl;
    cin>>a>>b>>c;
    p=(a+b+c)/2;
    s=sqrt((p-a)*(p-b)*(p-c)*p);
    r=2*s/(a+b+c);
    cout<<"r="<<r;
    return 0;
}
```



```
uchburchak tomonlarini kiriting
5
6
4
r=1.32288
```

2. Tomonlar uzunligi a , b , c , d ga teng bo‘lgan to‘rtburchakka tashqi chizilgan aylana radiusini hisoblash dasturini tuzing.

```
#include <iostream>
```

```

#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int a,b,c,d;
    float p,S,R;
    cout<<"To'rtburchak tomonlarini kiriting" << endl;
    cin >> a >> b >> c >> d;
    p = (a+b+c+d)/2;
    S = sqrt((p-a)*(p-b)*(p-c)*(p-d));
    R = sqrt((a*b+c*d)*(a*c+b*d)*(a*d+b*c))/(4*S);
    cout << "To'rtburchakka tashqi chizilgan aylana ";
    cout << "R="<<R;
    return 0;
}

```

```

To'rtburchak tomonlarini kiriting
6
5
8
9
To'rtburchakka tashqi chizilgan aylana R=5.07947

```

3. Muntazam n burchakli ko'pburchakka tashqi chizilgan R aylana radiusi orqali uning a tomonni hisoblash dasturini tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int R,n;
    float a;
    cout<<"R va n ni kiriting" << endl ;
    cin>>R>>n;
    a=2*R*sin(3.14/n);
    cout<<"a="<<a;
    return 0;
}

```

```

R va n ni kiriting
5
6
a=4.9977

```

4. Muntazam n burchakli ko'pburchakka tashqi chizilgan R aylana radiusi orqali unga ichki chizilgan aylana radiusini hisoblash dasturini tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int R,n;
    float r1;
    cout<<"R va n ni kiriting" << endl;
    cin>>R>>n;
    r1=R*cos(3.14/n);
    cout<<"r1="<<r1;
    return 0;
}

```

```

R va n ni kiriting
9
4
r1=6.36649

```

5. To'g'ri to'rtburchakning perimetrini hisoblaydigan dastur tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float a,b,p;
    cin>>a;
    cin>>b;
    p=2*(a+b);
    cout<<"p="<<p;
    return 0;
}

```

```

6
8
p=28

```

6. Trapetsiya yuzini hisoblaydigan dastur tuzing.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    float a,b,h,S;
    cin>>a>>b>>h;
    S=(a+b)*h/2;
    cout<<"S="<<S;
}

```

```
return 0;
}
```

```
6
5
4
$=22
```

7. R radiusli doira berilgan. Uni chegaralagan aylana yuzi va uzunligini aniqlaydigan dasturni tuzing.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float R,l,S;
    float const pi=3.14;
    cin>>R;
    l=2*pi*R;
    S=pi*R*R;
    cout<<"l="<<l;
    cout<<"S="<<S;
    return 0; }
```

```
6
l=37.68 S=113.04
```

8. a, b, c tomonli uchburchak balandliklari uzunligini aniqlaydigan dastur tuzing.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float R,l,S;
    float const pi=3.14;
    cin>>R;
    l=2*pi*R;
    S=pi*R*R;
    cout<<"l="<<l<<"\t"<<"S="<<S;
    return 0;
}
```

```
6
l=37.68 S=113.04
```

9. a, b, c tomonli uchburchak medianalari uzunligini aniqlaydigan dastur tuzing.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float a,b,c;
    float ma,mb,mc;
    cin>>a>>b>>c;
    ma=0.5*sqrt(2*(b*b+c*c)-a*a);
    mb=0.5*sqrt(2*(a*a+c*c)-b*b);
    mc=0.5*sqrt(2*(b*b+a*a)-c*c);
    cout<<"ma="<<ma;
    cout<<"mb="<<mb;
    cout<<"mc="<<mc;
    return 0;
}
```

```
6
4
8
ma=5.56776      mb=6.78233      mc=3.16228
```

10. a va b katetlari bo'yicha to'g'ri burchakli uchburchak yuzini hisoblaydigan dastur tuzing.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int a,b;
    float S;
    cin>>a>>b;
    S=a*b/2;
    cout<<"S="<<S;
    return 0;
}
```

```
5
6
S=15
```

11. H balandlikka M massali jismni yuqoriga ko'targanda bajarilgan ishni hisoblovchi dasturni tuzing. Erkin tushish tezlanishini o'zgarmas $G=9,81$ deb oling.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float M,H,A;
    float const G=9.81;
    cin>>M>>H;
    A=M*G*H;
    cout<<"A="<<A;
    return 0;
}

```

```

6
4
A=235.44

```

12. R zichligi va H balandlikli suyuqlik ustunini idish tubiga beradigan bosimini aniqlaydigan dasturni tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float R,H,P;
    float const G=9.81;
    cin>>R>>H;
    P=R*G*H;
    cout<<"P="<<P;
    return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
5
6
P=294.3Для продолжения нажмите любую клавишу . . .

```

13. R zichlikli suyuqlikka yarim botirilgan V hajmli jismga ta'sir etuvchi siqib chiqarish kuchini aniqlaydigan dastur tuzing.

```

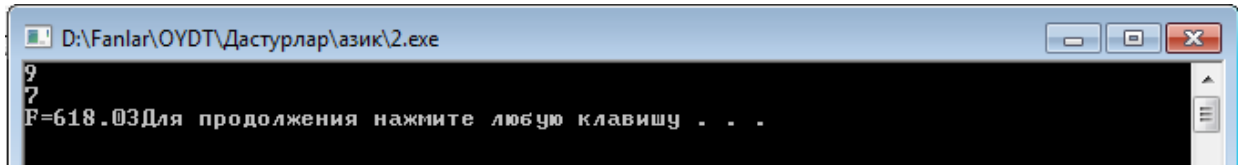
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float R,V,F;
    float const G=9.81;

```

```

cin>>R>>V;
F=R*G*V;
cout<<"F="<<F;
return 0;
}

```

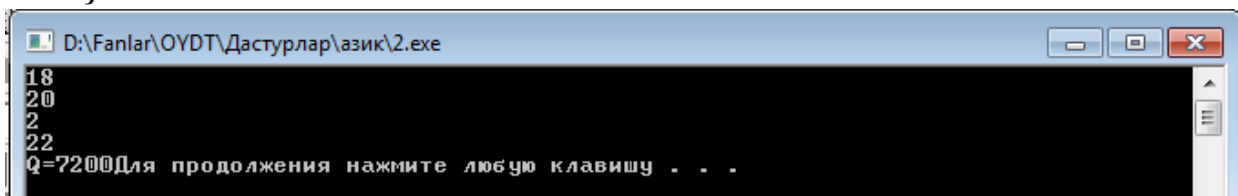


14. $Q=cm(t_2-t_1)$ formula bo'yicha issiqlik miqdorini aniqlaydigan dastur tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float c,m,t1,t2,Q;
    cin>>c>>m>>t1>>t2;
    Q=c*m*(t2-t1);
    cout<<"Q="<<Q;
    return 0;
}

```



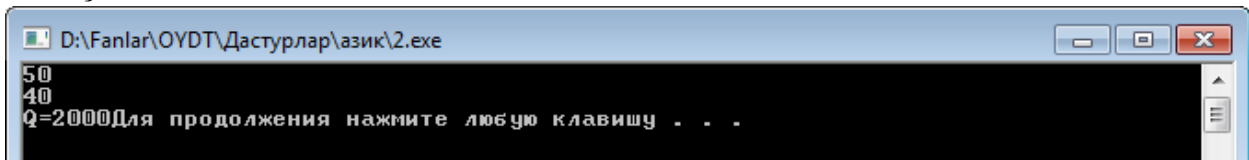
15. q yonish issiqlikka ega bo'lgan M massali kerosinning yonishi natijasida olingan issiqlik miqdorini aniqlaydigan dastur tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float q1,M,Q;
    cin>>q1>>M;
    Q=q1*M;
    cout<<"Q="<<Q;
    return 0;
}

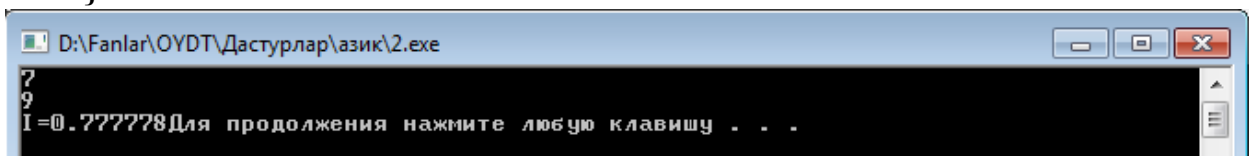
```

```
}
```



16. U kuchlanishli, R qarshilikli elektr zanjirida to'k kuchini aniqlaydigan dasturni tuzing.

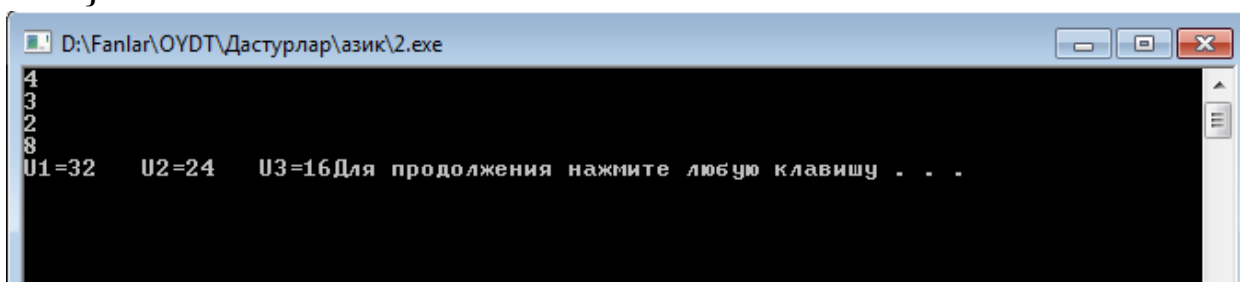
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    float R,U,I;
    cin>>U>>R;
    I=U/R;
    cout<<"I="<<I;
    return 0;
}
```



17. R_1, R_2, R_3 qarshilikli ketma-ket ulangan elektr zanjirida kuchlanishni aniqlaydigan dastur tuzing. Bunda U kuchlanishda tok kuchi I ni tashkil etadi.

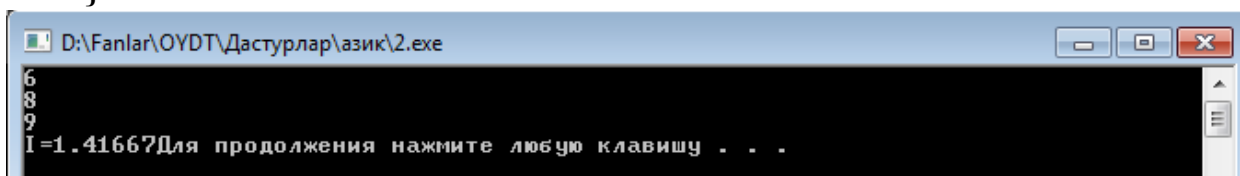
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    float R1,R2,R3,I;
    float U1,U2,U3;
    cin>>R1>>R2>>R3>>I;
    U1=I*R1;
    U2=I*R2;
    U3=I*R3;
    cout<<"U1="<<U1<<"\t"<<"U2="<<U2<<"\t"<<"U3="<<U3;
    return 0;
}
```

}



18. R_1 va R_2 qarshilikli parallel ulangan 2 ta rezistorning tok kuchini aniqlang. Bu rezistorlarning oxiridagi kuchlanish U ga teng.

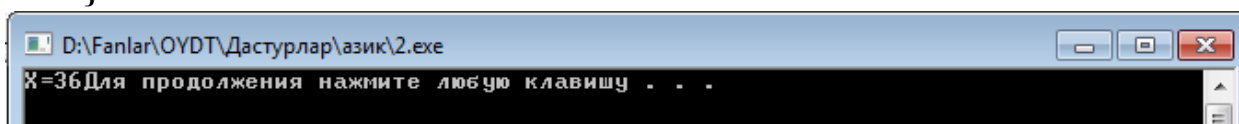
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    float U,R1,R2,I1,I2,I;
    cin>>U>>R1>>R2;
    I1=U/R1;
    I2=U/R2;
    I=I1+I2;
    cout<<"I="<<I;
    return 0;
}
```



19. Bug'doy urug'ning o'sib chiqish ko'rsatkichi 95% ga teng. Ekilgan bug'doyning 35 tonnasi o'sib chiqish uchun necha tonna bug'doy ekish kerakligi to'g'risida dastur tuzing.

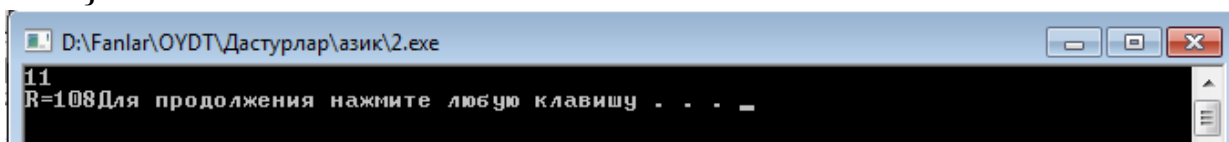
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int V,P;
    float X;
    V=35;
    P=95;
    X=V*100/P;
    cout<<"X="<<X;
    return 0;
}
```

}



20. Komil va Odil yashaydigan uylar orasidagi masofa 1200 m. Bir kuni ular uylaridan chiqib bir-biriga qarab yo‘l olishdi. Komil A metr masofani o‘tganda uchrashishibdi. Odil o‘tgan masofa Komil o‘tgan masofadan necha barobar katta?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int A,B;
    float const C=1200;
    float R;
    cin>>A;
    B=C-A;
    R=B/A;
    cout<<"R="<<R;
    return 0;
}
```



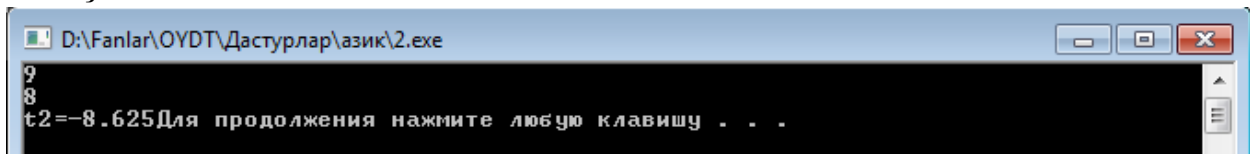
21. Bekatdan ko‘lgacha bo‘lgan masofa S km. Turistlar bekatdan ko‘lga qarab yo‘l olishdi, ular bir yarim soat A km/soat tezlik bilan yayov yurishdi so‘ngra ular B km/soat tezlik bilan ketayotgan mashinaga o‘tirishibdi. Turistlar qancha vaqtda ko‘lga yetib borishadi.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float const S=3;
    float t1,v1,v2,S1,S2,t2;
    cin>>v1>>v2;
    S1=v1*v2;
    S2=S-S1;
    t2=S2/v2;
```

```

cout<<"t2="<<t2;
return 0;
}

```

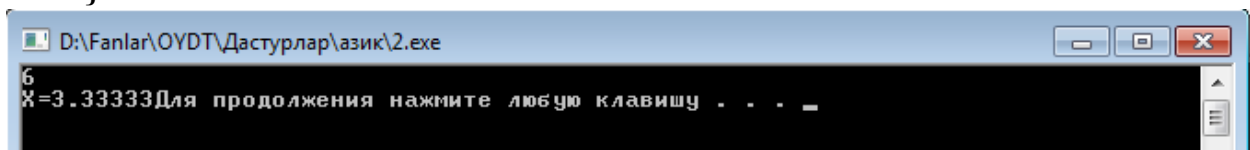


22. 18 t temir rudasidan 10 t temir eritib olishadi. A t rudadan qancha temir eritib olish mumkin?

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float A,X;
    cin>>A;
    X=10*A/18;
    cout<<"X="<<X;
    return 0;
}

```

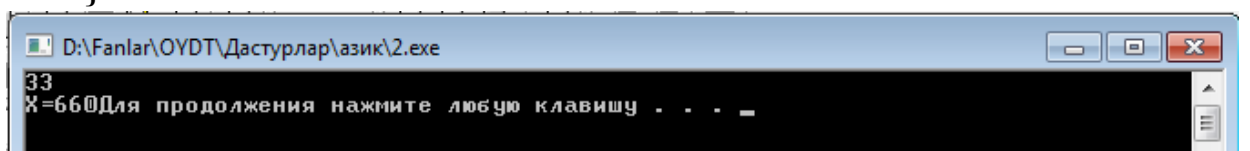


23. 1,5 l choyni damlash uchun 30 g quruq choy kerak bo'ladi. Choynak o'ziga B l sig'iradi. Damlash uchun qancha quruq choy kerak?

```

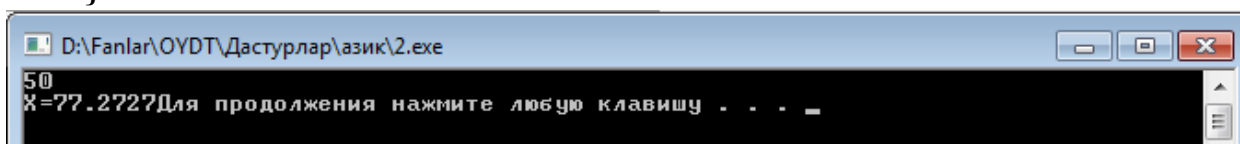
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float B,X;
    cin>>B;
    X=30*B/1.5;
    cout<<"X="<<X;
    return 0;
}

```



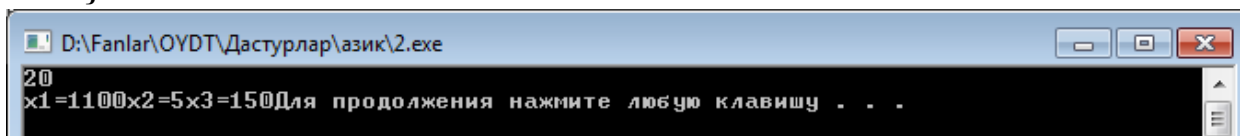
24. 56 m uzunlikka ega bo'lgan maktab koridorining polini bo'yab chiqish kerak. 22 m uzunlikli koridorning bir qismini bo'yab chiqish natijasida A kg bo'yoq sarf bo'ldi. Koridorni oxirigacha bo'yab chiqish uchun yana qancha bo'yoq kerak bo'ladi.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    float A,X;
    cin>>A;
    X=(56-22)*A/22;
    cout<<"X="<<X;
    return 0;
}
```



25. To'rt porsiya guruchli bo'tqani tayyorlash uchun 220 g guruch, 1 l sut va 30 g shakar olish kerak. N porsiyali bo'tqa tayyorlash uchun bu masalliqlardan qancha olish kerak.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int N;
    float x1,x2,x3;
    cin>>N;
    x1=220*N/4; x2=N/4; x3=30*N/4;
    cout<<"x1="<<x1<<"x2="<<x2<<"x3="<<x3;
    return 0;
}
```



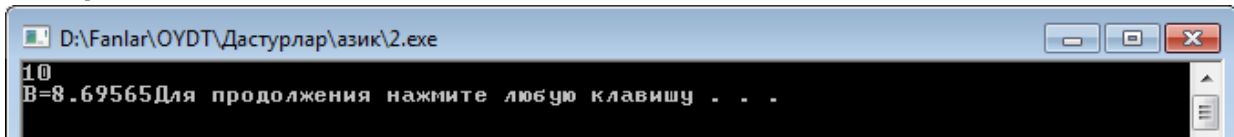
26. Univermark noyabrdan A mln so'mlik tovarni sotdi bu esa oktabrdagi ko'rsatkichdan 15% ko'proq. Oktabrda tovarlar necha pulga sotilgan.

```
#include <iostream>
```

```

using namespace std;
int main(int argc, char *argv[]){
    float A,B;
    cin>>A;
    B=A/(1.15);
    cout<<"B="<<B;
    return 0;
}

```

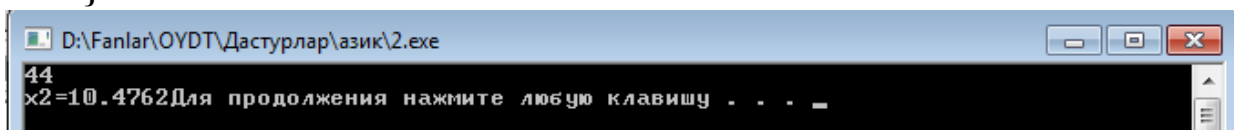


27. Shakarli qizilcha tarkibida 14% shakar bor. 1 ga dan shakarli qizilcha yig'ishadi. B t shakar olish uchun qancha gektar yerga shakarli qizilcha ekish kerak.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    float B,x1,x2;
    cin>>B;
    x1=100*B/14;
    x2=x1/30;
    cout<<"x2="<<x2;
    return 0;
}

```



28. Masqaraboz davrada o'tirganlarning har biriga son o'ylashni aytdi. So'ngra shunday dedi "O'ylagan songa 5 ni qo'shing. Endi natijadan 2 ni ayiring. So'ngra natijaga 7 ni qo'shing". So'ngra masqaraboz xohlovchilarning har birida qanday son hosil bo'lganini so'radi. Javobni eshitib u o'ylab ham turmay, ular o'ylagan sonni e'lon qiladi. Masqsaraboz fokusini takrorlaydigan dasturni tuzing.

```

#include<iostream>
using namespace std;
int main(){
    int a,x;

```

```

cout<<endl;
cout<<"-Bir son o'ylang\n";
cout<<"-Shu songa beshni qo'shing\n";
cout<<"-Hosil bo'lgan sondan ikkini ayiring\n";
cout<<"-Endi hosil bo'lgan songa yettini qo'shing\n";
cout<<"-Hosil bo'lgan sonni ayting: ";
cin>>a;
x=a-10;
cout<<"Siz o'ylagan son "<<x<<" ga teng "<<endl;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
-Bir son o'ylang
-Shu songa beshni qo'shing
-Hosil bo'lgan sondan ikkini ayiring
-Endi hosil bo'lgan songa yettini qo'shing
-Hosil bo'lgan sonni ayting: 20
Siz o'ylagan son 10 ga teng
Для продолжения нажмите любую клавишу . . .

```

29. Pechenya og'izda eriydigan bo'lishi uchun unga maxsus poroshok solishadi. Uning tarkibida 25% ozuqa sodasi 15 % limon kislotasi va 60% un kiradi. A gr maxsus poroshok tayyorlash uchun qancha soda, limon kislotasi va un olish kerak.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float A,x1,x2,x3;
    cin>>A;
    x1=A*25/100;
    x2=A*15/100;
    x3=A*40/100;
    cout<<x1<<x2<<x3;
    return 0;
}

```

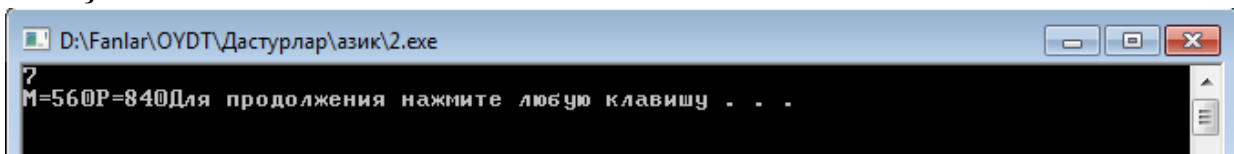
```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
55
13.758.2522Для продолжения нажмите любую клавишу . . .

```

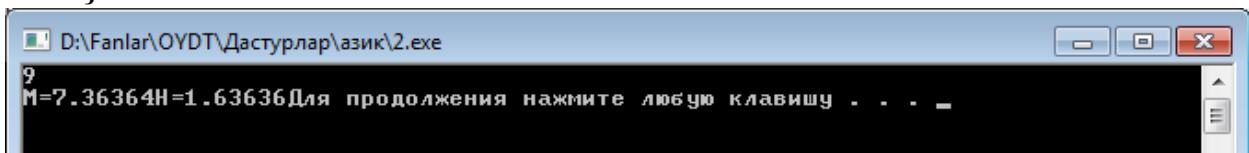
30. Bitta sharf to‘qish uchun 200 gr to‘qish uchun mo‘ljallangan ip kerak. Bu ip tarkibida 40% moxera 60% poliakril, bor. B ta sharflar tayyorlash uchun qancha moxser va poliakril kerak bo‘ladi.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
    float B,X,M,P;
    cin>>B;
    X=200*B; M=40*X/100; P=60*X/100;
    cout<<"M="<<M<<"P="<<P;
    return 0;
}
```



31. Melxior –bu nikel va rux ning aralashmasi, ularning 2 va 9 sonlariga proporsional. C kg li melxior tayyorlash uchun qancha nikel va rux kerak bo‘ladi.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
    float C,M,H;
    cin>>C;
    M=9*C/11; H=2*C/11;
    cout<<"M="<<M<<"H="<<H;
    return 0;
}
```



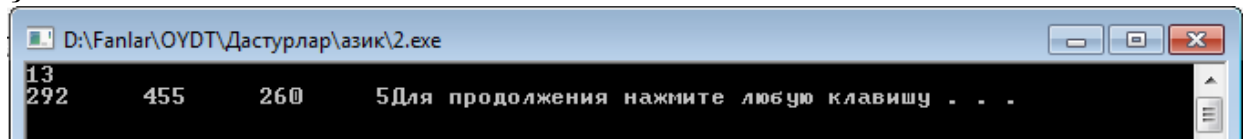
32. Tuzlangan karam tayyorlash uchun 10 kg karamga 225 g tuz, 350 gr sabzi 200 gr klyukva 4 gr barg olishadi. M kg karam sotib olingan bo‘lsa qancha tuz, sabzi, klyukva va barg olish kerak.

```
#include <iostream>
using namespace std;
```

```

int main(int argc, char *argv[]){
    int M;
    float x1,x2,x3,x4;
    cin>>M;
    x1=225*M/10; x2=350*M/10;
    x3=200*M/10; x4=4*M/10;
    cout<<x1<<"\t"<<x2<<"\t"<<x3<<"\t"<<x4;
    return 0;
}

```

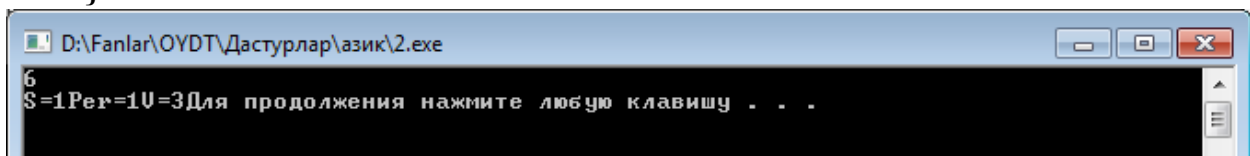


33. Shaftolidan kompot tayyorlash uchun 1:1:3 proporsiyada shakar, shaftoli va suv solishadi. P kg kompot tayyorlash uchun har bir mahsulotdan qancha kg olish kerak.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int P; float S,Per,V;
    cin>>P;
    S=P/5;
    Per=P/5;
    V=3*P/5;
    cout<<"S="<<S<<"Per="<<Per;
    cout<<"V="<<V;
    return 0;
}

```



34. Odam organizmining, uning massasidan 65% ini kislorod molekulalarining massalari tashkil etadi. Odam organizmidagi kislorod molekulalarining massalarini aniqlaydigan dastur tuzing.

```

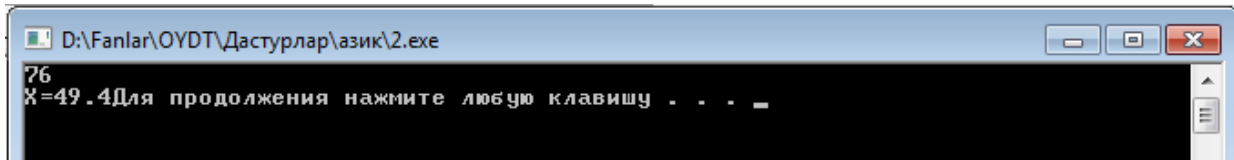
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){

```

```

float M,X;
cin>>M;
X=65*M/100;
cout<<"X="<<X;
return 0;
}

```

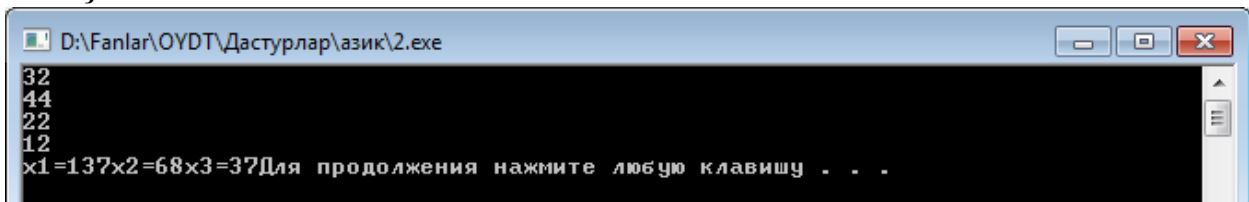


35. Sinfda N ta o'quvchi bor. Yozma ishdan so'ng A ta besh B ta to'rt C ta uch olindi .Uchlar, to'rtlar va beshlar protsentlarini toping.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int N,A,B,C;
    float x1,x2,x3;
    cin>>N>>A>>B>>C;
    x1=100*A/N;
    x2=100*B/N;
    x3=100*C/N;
    cout<<"x1="<<x1;
    cout<<"x2="<<x2;
    cout<<"x3="<<x3;
    return 0;
}

```



36. To'rtta sinfga F ga teng yuzali maydonni yig'shtirish buyirildi. Har bir sinfdagi o'quvchilar soni ma'lum. Har bir sinf qancha yuzaga teng maydonni yig'ishtirish kerak?

```

#include <iostream>
#include <math.h>
using namespace std;

```

```

int main(int argc, char *argv[]){
    float N1,N2,N3,N4,F,N;
    float F1,F2,F3,F4;
    cin>>N1>>N2>>N3>>N4>>F;
    F1=F*N1/N;
    F2=F*N2/N;
    F3=F*N3/N;
    F4=F*N4/N;
    cout<<"F1="<<F1<<"F2="<<F2<<"F3="<<F3<<"F4="<<F4;
    return 0;
}

```

37. Ikkita son berilgan. Bu sonlar kublarining o‘rta arifmetigi va bu sonlar modullarining o‘rta geometrigi topilsin.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int a,b;
    float c,d;
    cin>>a;
    cin>>b;
    c=(a*a*a+b*b*b)/2;
    d=sqrt(abs(a)*abs(b));
    cout<<"c="<<c<<"d="<<d;
    return 0;
}

```

38. x_1 , y_1 va x_2 , y_2 koordinatali 2 ta nuqtalar orasidagi masofani toping.

```

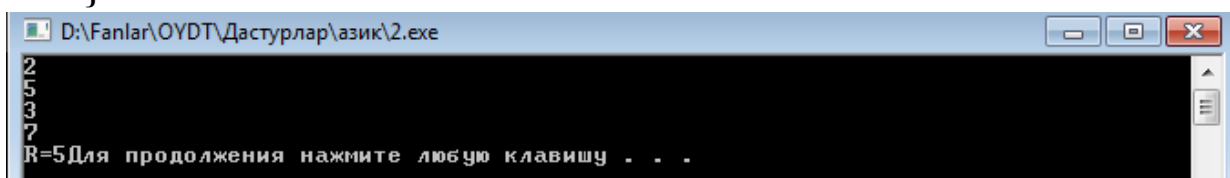
#include <iostream>

```

```

#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int x1,x2,y1,y2;
float R;
cin>>x1;
cin>>x2;
cin>>y1;
cin>>y2;
R=sqrt((x2-x1)*(x2-x1)+(y2-y1)*(y2-y1));
cout<<"R="<<R;
return 0;
}

```

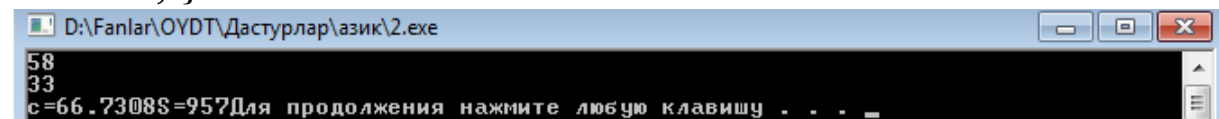


39. Berilgan 2 ta katetlar orqali to'g'ri burchakli uchburchakning gipotenuza va yuzini hisoblab toping.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float a,b,c,S;
cin>>a>>b;
c=sqrt(a*a+b*b);
S=a*b/2;
cout<<"c="<<c<<"S="<<S;
return 0; }

```



40. x va y sonlar berilgan. Ularning yig'indisini, ayirmasini va ko'paytmasini toping .

```

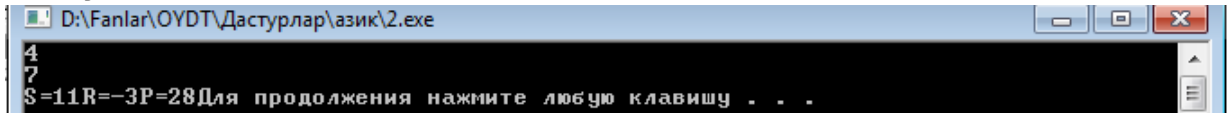
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){

```

```

int x,y,S,R,P;
cin>>x>>y;
S=x+y;
R=x-y;
P=x*y;
cout<<"S="<<S<<"R="<<R<<"P="<<P;
return 0;
}

```

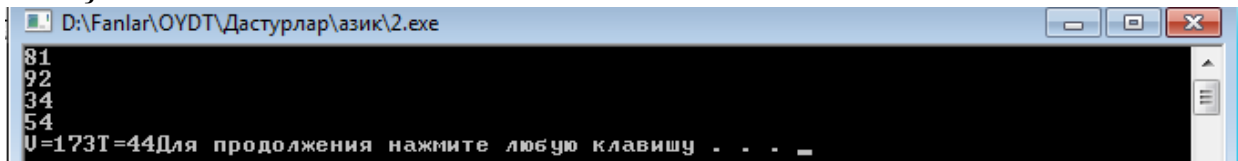


41. t_1 temperaturali v_1 litr t_2 temperaturali v_2 litr suv bilan aralashtirilgan. Hosil bo'lgan aralashmaning hajmi va temperaturasini hisoblab toping.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float V1,V2,T1,T2,V,T;
cin>>V1>>V2>>T1>>T2;
V=V1+V2;
T=(T1+T2)/2;
cout<<"V="<<V<<"T="<<T;
return 0;
}

```



42. m gr massali $w\%$ li eritmani tayyorlash uchun kerak bo'ladigan tuz va suv massasini aniqlaydigan dastur tuzing.

```

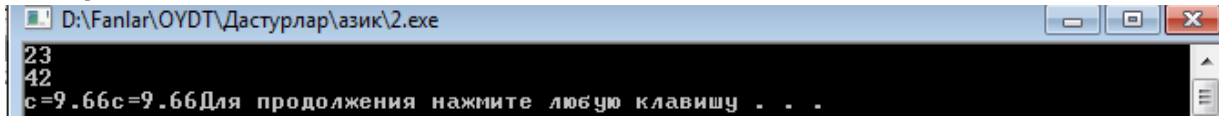
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float c,V,w,m;
cin>>w>>m;
c=(w*m)/100;
V=m-c;
}

```

```

cout<<"c="<<c<<"V="<<V;
return 0;
}

```

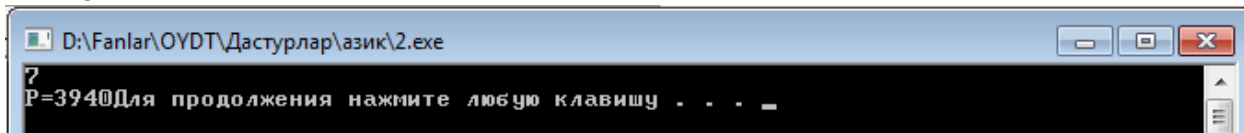


43. x berilgan $2x^4 - 3x^2 + 4x^3 - 5x + 6$ ni hisoblang. Operatsiyani kam ishlatish haqida ham o'ylang.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float x,P;
cin>>x;
P=2;
P=P*x-3;
P=P*x+4;
P=P*x-5;
P=P*x+6;
cout<<"P="<<P;
return 0;
}

```



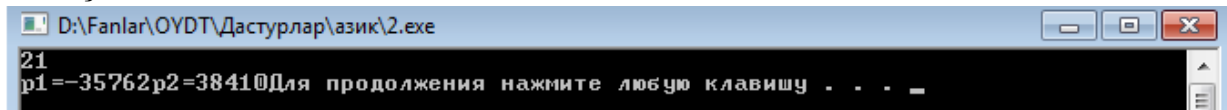
44. x berilgan $1-2x+3x^2-4x^3$ va $1+2x+3x^2+4x^3$ ni hisoblang. Operatsiyani kam ishlatish haqida ham o'ylang.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float x,p1,p2;
cin>>x;
p1=-4; p1=p1*x+3; p1=p1*x-2; p1=p1*x+1;
p2=4; p2=p2*x+3; p2=p2*x+2; p2=p2*x+1;
cout<<"p1="<<p1<<"p2="<<p2;
return 0;
}

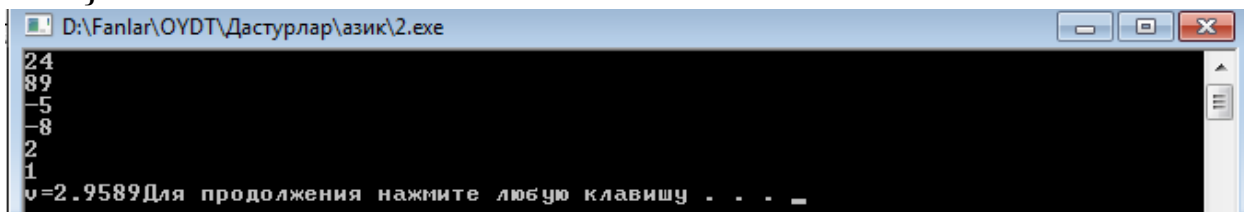
```

```
}
```



45. Avtomobil yo'lining 3 ta har xil uzunlikli qismini turli xil tezliklar bilan o'tdi. Avtomobilning o'rtacha tezligini aniqlang.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float s1,s2,s3,v1,v2,v3,t1,t2,t3;
float s,t,v;
cin>>s1>>s2>>s3>>v1>>v2>>v3;
t1=s1/v1;
t2=s2/v2;
t3=s3/v3;
s=s1+s2+s3;
t=t1+t2+t3;
v=s/t;
cout<<"v="<<v;
return 0;
}
```



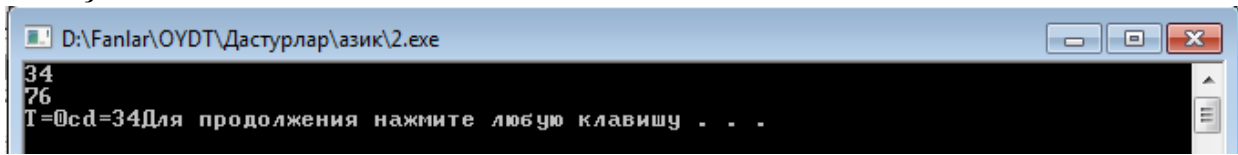
46. Xaridorda mavjud bo'lgan pul miqdori va bitta tovarning narxi ma'lum. Xaridor qancha tovar sotib olishi mumkin va uning qaytimi qancha?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int S,c,T;
float cd;
cin>>S>>c;
T=S / c;
cd=S % c;
```

```

cout<<"T="<<T;
cout<<"cd="<<cd;
return 0;
}

```

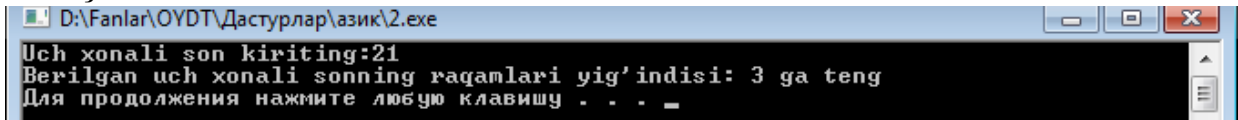


47. Klaviaturadan kiritilgan uch xonali natural sonning raqamlari yig'indisini aniqlaydigan dastur tuzing.

```

#include<iostream>
using namespace std;
int main(){
int n,k,d,s;
cout<<"Uch xonali son kiriting:";
cin>>n;
k=n%10; n=(n-k)/10; d=n%10; n=(n-d)/10;
s=k+n+d;
cout<<"Berilgan uch xonali sonning raqamlari yig'indisi:
"<<s<<" ga teng"<<endl;
return 0;
}

```

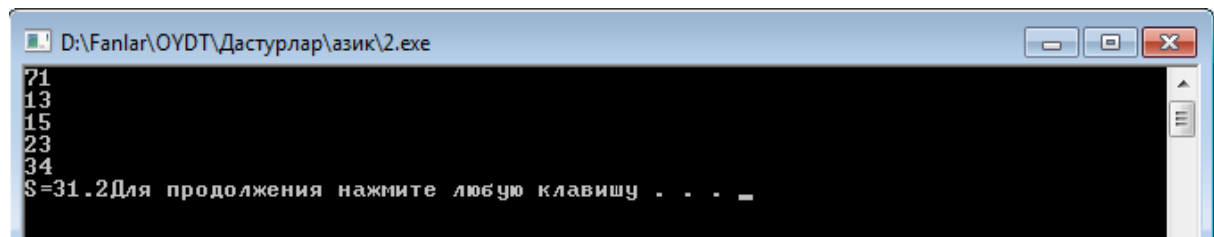


48. Beshta korxonada chiqarilayotgan mahsulot hajmi ma'lum. Bitta korxona tomonidan ishlab chiqarilayotgan mahsulotning o'rtacha hajmi hisoblansin.

```

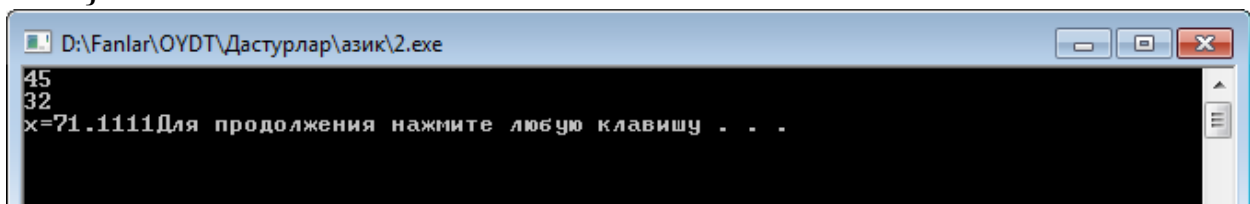
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a1,a2,a3,a4,a5;
float S;
cin>>a1>>a2>>a3>>a4>>a5;
S=(a1+a2+a3+a4+a5)/5;
cout<<"S="<<S;
return 0;}

```



49. Korxona tomonidan ishlab chiqarilayotgan mahsulotlarning rejaviy P va amaliy F ko'rsatkichi ma'lum. Rejaning bajarilish prosentini aniqlang.

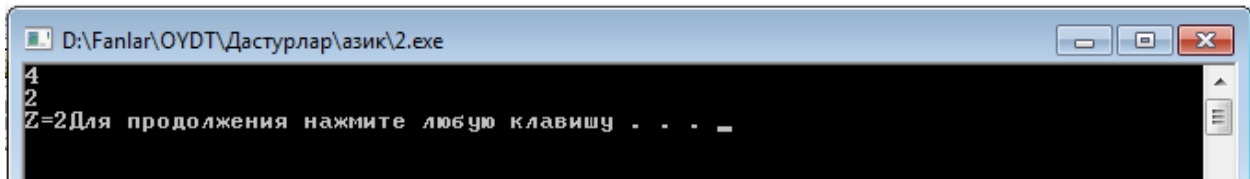
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float FF,PF,x;
cin>>PF>>FF;
x=FF*100/PF;
cout<<"x="<<x;
return 0;
}
```



50. Qurilish ishlarida N ta quruvchi S soat ishlab berish zarur. Bitta odamning o'rtacha yuklanganligi qancha?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
int S,N;
float Z;
cin>>S>>N;
Z=S/N;
cout<<"Z="<<Z;
return 0;
}
```

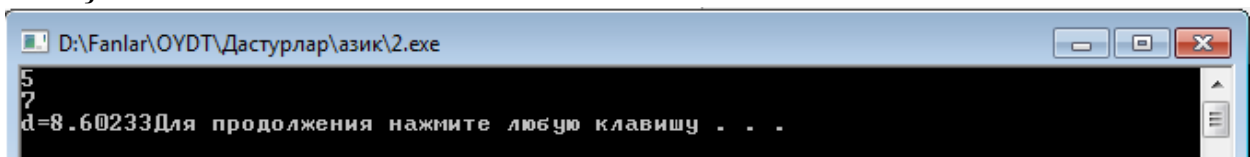
}



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
4
2
Z=2Для продолжения нажмите любую клавишу . . .
```

51. Koordinata boshidan chiquvchi 2 ta vektor oxirlarining koordinatalari berilgan. Bu vektor uzunliklari topilsin.

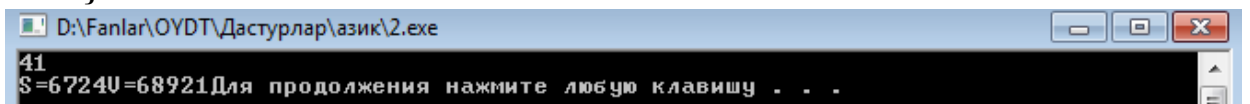
```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
float x,y,d;
cin>>x>>y;
d=sqrt(x*x+y*y);
cout<<"d="<<d;
return 0;
}
```



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
5
7
d=8.60233Для продолжения нажмите любую клавишу . . .
```

52. Kub qirrasi uzunligi berilgan. Kub hajmi va yon sirlari yuzasi topilsin.

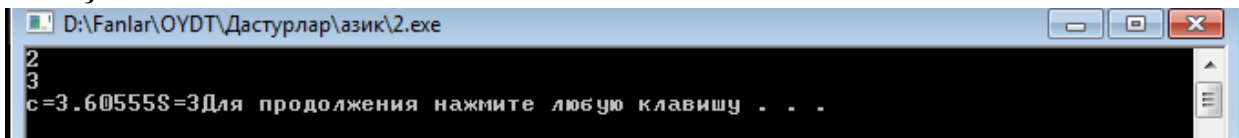
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a,b,S,V;
cin>>a;
b=a;
a=a*a;
S=4*a;
V=a*b;
cout<<"S="<<S<<"V="<<V;
return 0;
}
```



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
41
S=6724V=68921Для продолжения нажмите любую клавишу . . .
```

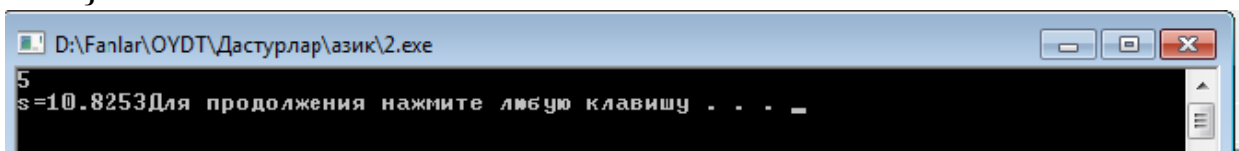
53. To'g'ri burchakli uchburchakning katetlari berilgan. Uning gepotenuzasi va yuzasi topilsin.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float a,b,c,S;
    cin>>a>>b;
    c=sqrt(pow(a,2)+pow(b,2));
    S=a*b/2;
    cout<<"c="<<c<<"S="<<S;
    return 0;
}
```



54. Teng tomonli uchburchak tomoni berilgan. Uning yuzasi topilsin.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    float a,s;
    cin>>a;
    s=sqrt(3.0)*a*a/4;
    cout<<"s="<<s;
    return 0;
}
```



55. 1 kv metr matoni ishlab chiqish uchun 5 minut vaqt sarf qilinadi. L kenglikli matodan N soatda qancha ishlab chiqish mumkin.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
```

```

int n;
float l,d;
cin>>n>>l;
d=n*60/(5*l);
cout<<"d="<<d;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
3
d=36Для продолжения нажмите любую клавишу . . .

```

56. Sun'iy iplarni ishlab chiqish uchun xom ashyo R radiusli va H balandlikli bochkalarda yetkaziladi. N bochkalarning hajmi aniqlansin.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float const pi=3.14;
float R,V;
int H,N;
cin>>N>>H>>R;
V=pi*N*R*R*H;
cout<<"V="<<V;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
5
6
8
V=6028.8Для продолжения нажмите любую клавишу . . .

```

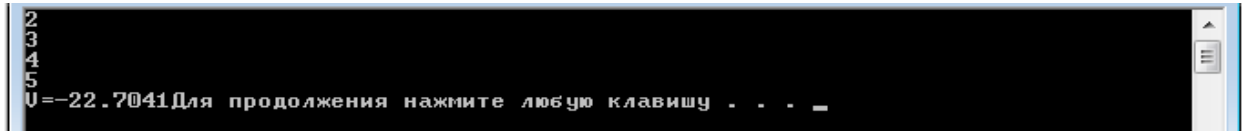
57. Asosning ikki tomoni ular orasidagi burchagi va balandligi bo'yicha parallapiped hajmi aniqlansin .

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float a,b,alfa,V;
int H;

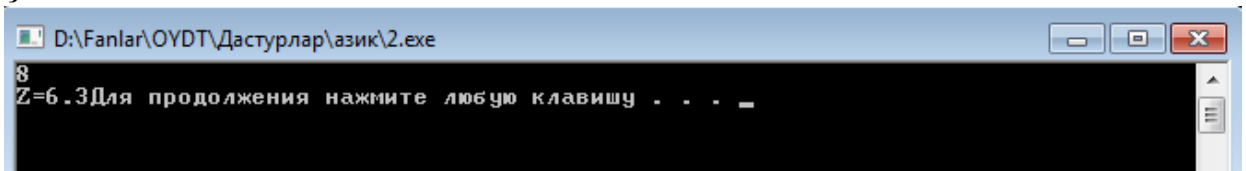
```

```
cin>>a>>b>>alfa>>H;
V=a*b*sin(alfa)*H;
cout<<"V="<<V;
return 0; }
```



58. Asbob uskunalar tuzadigan sex har kuni P foyda oladi, ulardan 70% boj to‘laydi. Foydaning 30% qolgan qismi ishlab chiqarishning rivojlanishiga, qolgani esa 8 ta ishchining maoshiga sarf bo‘ladi. Ishchilarning o‘rtacha maoshi qancha?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float P,Z;
cin>>P;
Z=(P-0.7*P-0.3*(P-0.7*P))*30/8;
cout<<"Z="<<Z;
return 0;
}
```



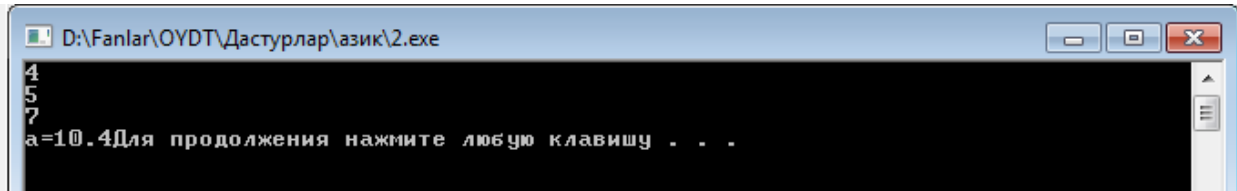
59. Shadrin o‘rdagi quyidagicha tayyorlanadi. O‘rdak go‘shhini yuvib va salfetka bilan ho‘llab, tuz va garmdori bilan artiladi. 300 g guruchni qaynoq suvga 10 min tindirib qo‘yamiz, so‘ngra suvni to‘kib yuboramiz. 700 g nordon olmani uzunchoq qilib o‘rtasini olib tashlab 0,7 sm dan kesamiz. O‘rdak go‘shhtining ichiga qavat-qavat qilib guruch va olmani solamiz. O‘rdakni qorni bilan yotqizib yaltiroq qog‘oz (folga) bilan yopamiz va o‘rtacha olovga 1-1,5 soatga qo‘yamiz. Gohida o‘rdak ustiga yog‘ quyib turamiz. Bu taom qancha turishini aniqlang.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
```

```

float a1,a2,a3,a;
cin>>a1>>a2>>a3;
a=a1+0.3*a2+0.7*a3;
cout<<"a="<<a;
return 0;
}

```

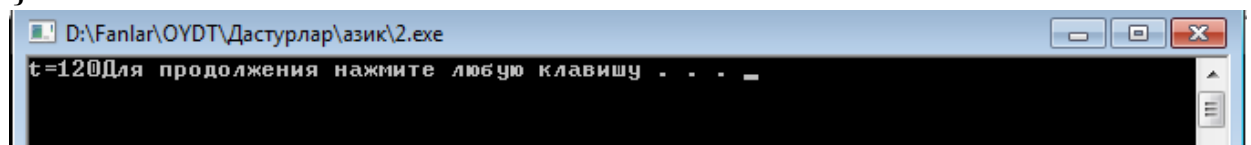


60. Militsiya mashinasi va jinoyatchi avtomobili orasidagi masofa 240 m. Militsiya mashinasining tezligi 40 m/s, jinoyatchi mashinasiniki esa 38 m/s. Qancha vaqtda militsiya xodimi jinoyatchini quvib yetadi.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float s,v1,v2,t;
v1=40;
v2=38;
s=240;
t=s/(v1-v2);
cout<<"t="<<t;
return 0;
}

```



61. O‘rtacha kattalikdagi temir kalit tarkibidagi atomlar sonini aniqlang. Ushbu atomlardan yaratilgan zanjir uzunligini toping.

```

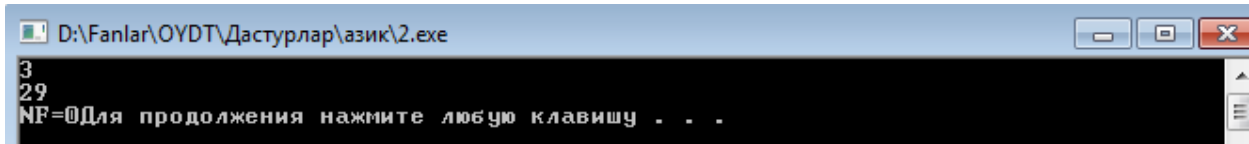
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float const A=6.02E23;
float n,NF;
int m,MR;

```

```

cin>>m>>MR;
n=m/MR;
NF=n*A;
cout<<"NF="<<NF;
return 0;
}

```

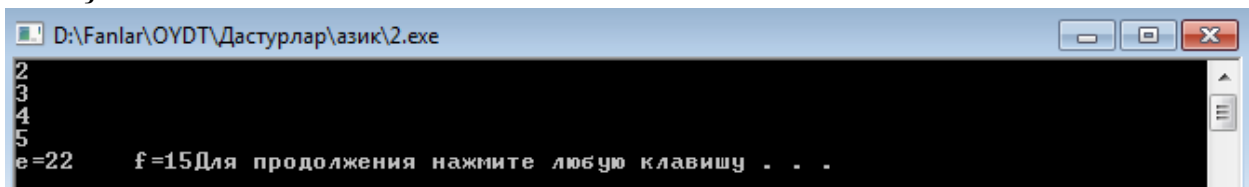


62. Ikkita oddiy kasrni qo‘shish dasturini tuzing.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int a,b,c,d,e,f;
cin>>a>>b>>c>>d;
e=a*d+c*b;
f=b*d;
cout<<"e="<<e<<"\t";
cout<<"f="<<f;
return 0;
}

```



63. Shamollashdan davolanish uchun 20 ta tabletka “Koldreks” yoki 30 ta tabletka “Aspirin Upsa” kerak bo‘ladi. Shaxsiy gigiyenaga rioya etish va chiniqish qoidalariga rioya etganda, tejalgan pullarga qancha Eskimo sotib olsa bo‘ladi?

```

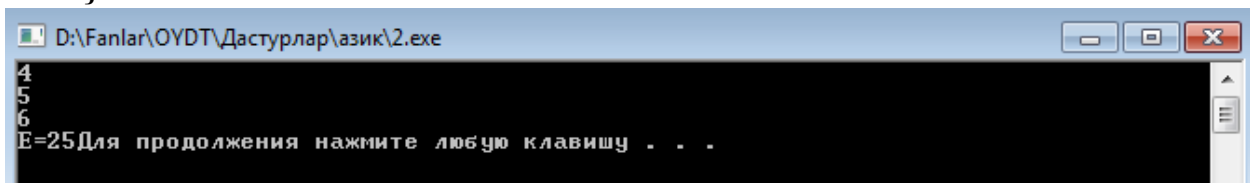
#include <iostream>
#include <math.h>
using namespace std;
int main(){
int PK,PA,PE;
float C,CK,CA;
float E,s,tv;

```

```

cin>>PK>>PA>>PE;
CK=20*PK;
CA=30*PA;
C=CK;
C=CA;
E=C/PE;
cout<<"E="<<E;
return 0;
}

```

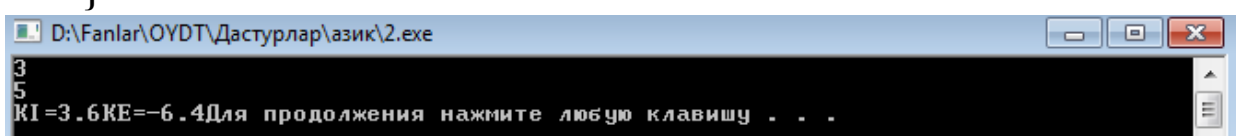


64. Xonaki tovuq tuxumini oʻrtacha radiusi 5 sm, inkubatorniki esa 4sm. Inkubator tuxumlarining barchasi xonaki tuxumlarning 10 tasiga mos keladi. Agar uy bekasi 10 ta xonaki tuxum olsa qancha soʻm tejaydi?

```

#include <iostream>
#include <math.h>
using namespace std;
int main(){
float const pi=3.14;
int R,R1;
float VD,VI,KI,KE;
cin>>R>>R1;
VD=4*pi*R*R;
VI=4*pi*R1*R1;
KI=10*VD/VI;
KE=(KI-10);
cout<<"KI="<<KI;
cout<<"KE="<<KE;
return 0;
}

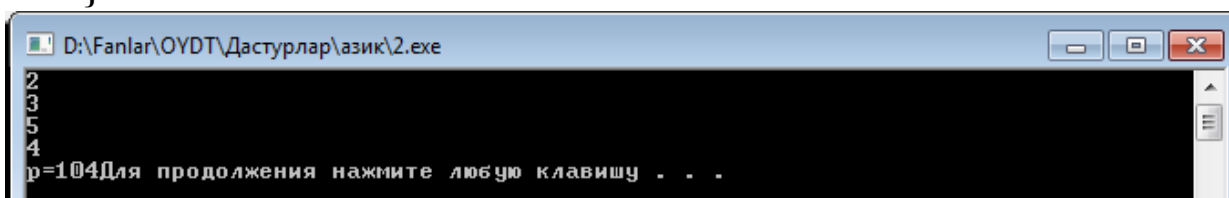
```



65. Ayol haftada uch juft bolalar, bir juft ayollar va bir juft erkaklar paypogʻini toʻqiydi, hamda ularni sotadi. Bir oyda toʻrt

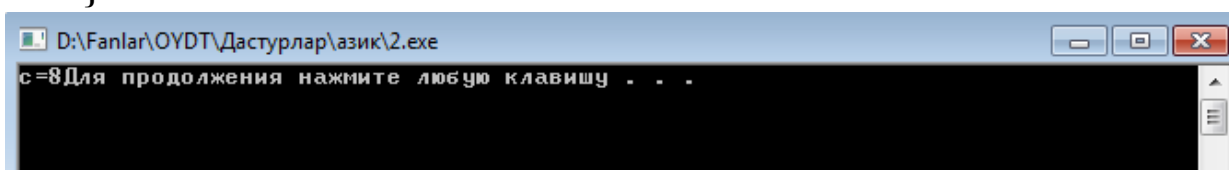
hafta borligini inobatga olib, ayol bir oyda qancha foyda ko‘rishini aniqlang?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int c1,c2,c3,seb,p;
cin>>c1>>c2>>c3>>seb;
p=4*(3*c1+2*c2+3*c3)-seb;
cout<<"p="<<p;
return 0;
}
```



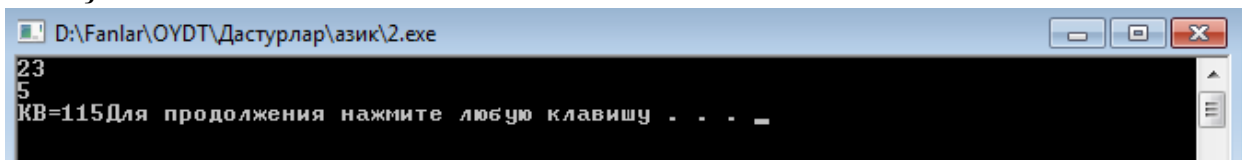
66. Bir yilda ijara to‘lovlari ikki marta oshadi. Birinchi marta 30% ga, ikkinchi marta esa 40% ga. Bir yil oldin Qudratullayevlar oilasi oyida 8 ming so‘m ijara to‘lovini to‘lashar edi. Hozir ular qancha to‘lashayapti?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int a,p1,p2;
float c;
a=8;
p1=30;
p2=40;
c=a*(1+p1/100)*(1+p2/100);
cout<<"c="<<c;
return 0;
}
```



67. Turistik vositalarini ishlab chiqaradigan sex bo‘limi, turistik palatkalar ishlab chiqaradi. Bo‘limda bir oylik rejani bajarish uchun kerak bo‘ladigan brezent miqdorini aniqlang.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int KP;
float KB,B;
cin>>KP>>B;
KB=KP*B;
cout<<"KB="<<KB;
return 0;
}
```



Masalani yechish uchun matematik model va dastur tuzish

68. Piyoda B_1 km ni T_1 soatda bosib o‘tadi. Piyoda T_2 soatda qancha masofani bosib o‘tadi?

Masalaning matematik modeli:

$T_1 - B_1$

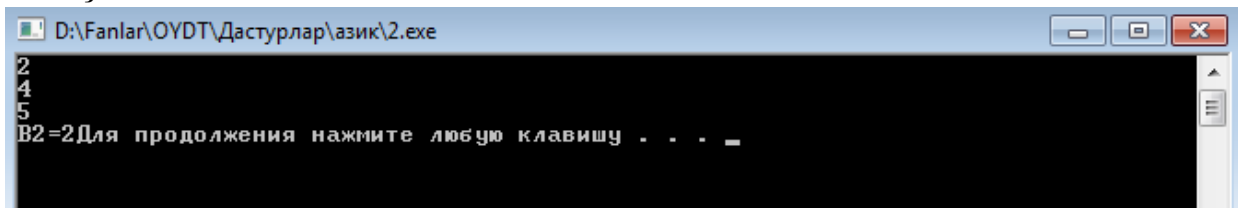
$$\Rightarrow B_2 = \frac{B_1 * T_2}{T_1}$$

$T_2 - B_2$

Masalaning dasturi:

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int B1,T1,T2;
float B2;
cin>>B1>>T1>>T2;
B2=B1*T2/T1;
cout<<"B2="<<B2;
```

```
return 0;
}
```



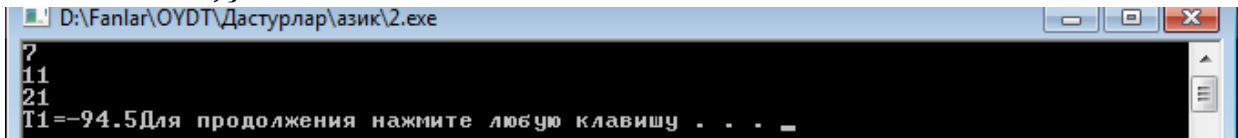
69. Motorli qayiq daryoda oqim bo‘ylab harakatlanib yo‘lni A punktdan B punktga T soatda bosib o‘tadi. Qancha vaqtda qayiq orqaga qaytadi?

Masalaning matematik modeli:

$$(v+v_1)T=(v-v_1)T_1 \quad \Leftrightarrow T_1=\frac{v+v_1}{v-v_1}T$$

Masalaning dasturi:

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float v,v1,T1,T;
cin>>v>>v1>>T;
T1=(v+v1)/(v-v1)*T;
cout<<"T1="<<T1;
return0;}
```



70. Gulqog‘ozning kengligi 70 sm. Xonadonini ta’mirlash uchun necha metr go‘lqog‘oz sotib olish kerak?

Masalaning matematik modeli:

$$S=2ab+2ac=2a(b+c)$$

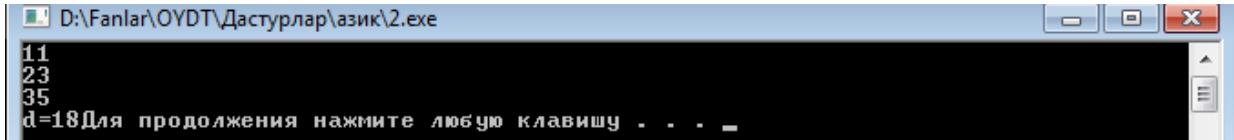
$$S=70*d$$

$$2a(b+c)=70*d \quad \Leftrightarrow a=\frac{70*d}{2(b+c)}$$

Masalaning dasturi:

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int a,b,c;
```

```
float d;
cin>>a>>b>>c;
d=2*a*(b+c)/70;
cout<<"d="<<d;
return0;}
```



A screenshot of a Windows command prompt window titled "D:\Fanlar\OYDT\Дастурлар\азик\2.exe". The window shows the execution of a C++ program. The output is "d=18". Below the output, there is a prompt in Russian: "Для продолжения нажмите любую клавишу . . .".

71. 1 m kv soxani bo'yash uchun x gr bo'yoq ketsa, stolni bo'yash uchun necha gr bo'yoq kerak bo'ladi?

Masalaning matematik modeli:

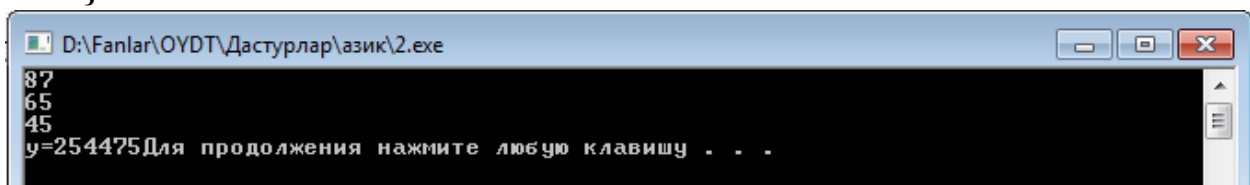
$$S=ab$$

$$1-x \quad y \Rightarrow \frac{S * x}{1} = S * x$$

S-y

Masalaning dasturi:

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int a,b,S,x;
    float y;
    cin>>a>>b>>x;
    S=a*b;
    y=S*x;
    cout<<"y="<<y;
    system("PAUSE");
    return 0;
}
```



A screenshot of a Windows command prompt window titled "D:\Fanlar\OYDT\Дастурлар\азик\2.exe". The window shows the execution of a C++ program. The output is "y=254475". Below the output, there is a prompt in Russian: "Для продолжения нажмите любую клавишу . . .".

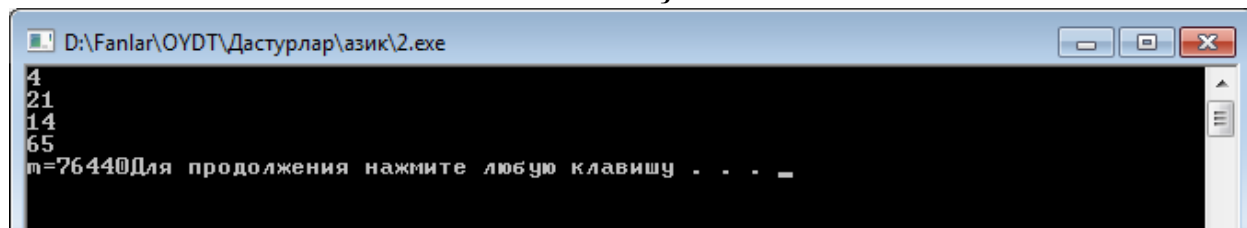
72. To'rtburchak shakldagi plastina massasini aniqlaydigan dastur tuzing.

Masalaning matematik modeli:

$$V=a*b*c; \quad m=p*V$$

Masalaning dasturi:

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a,b,c,v,m,ro;
cin>>a>>b>>c>>ro;
v=a*b*c;
m=ro*v;
cout<<"m="<<m;
system("PAUSE");
return 0;
}
```



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
4
21
14
65
m=76440Для продолжения нажмите любую клавишу . . . _
```

TARMOQLANUVCHI JARAYONLARNI DASTURLASH

Masalalarni dasturlashda, deyarli har doim, shartlarni qayta ishlash kerak bo'ladigan holatlarga duch kelamiz.

Masalan eng oson chiziqli tenglama $ax+b=0$ ni yechadigan va foydalanuvchi tomonidan oldindan so'raladigan a va b parametrlarni so'rovchi dastur tuzish kerak bo'lsin.

Aniq namoyon bo'ladiki, dasturda albatta, a nolga teng va a nolga teng bo'lmagan hollar ko'rib chiqilishi kerak, chunki bunday berilgan tenglama bitta ildizga ega bo'lishi yoki ildizga ega bo'lmasligi mumkin.

Agar $a=0$ bo'lsa, unda tenglama cheksiz ko'p yechimga ega bo'ladi. Bundan esa, x ning har qanday haqiqiy qiymati yechim bo'la olishi kelib chiqadi. Aks holda masala yechimga ega bo'la olmaydi.

Shartni tekshirish uchun bizga shartli operator kerak bo'ladi. Uni C++ dasturlash tilida ko'rib chiqaylik. C++ dasturlash tilida shartli operatorning shakli ikkita usulning birida ko'rsatiladi.

Shartli operatorlar

Shartli operatorlar biron bir shartning bajarilishiga qarab operatorlarning bajarilishi tartibini aniqlaydi. Bu yerda shart bu mantiqiy ifoda bo'lib uning bajarilishi natijasida true (chin) yoki false (yolg'on) natija hosil qilinadi va bu qiymatga qarab hisoblash jarayoni tarmoqlanadi.

if operatori umumiy holda quyidagicha yoziladi:

if (shart)

operator1;

else

operator2;

Agar shart chin qiymat qabul qilsa operator1 bajariladi, aks holda operator2 bajariladi.

if($x > y$)

$a=x$;

else $a=y$;

else konstruksiyasini yozmaslik ham mumkin:

```
if(x<0) x=-x;
```

```
abs=x;
```

Bu holda $x=-x$ operatori faqat x manfiy qiymatga ega bo'lsagina bajariladi. $abs=x$ operatori esa bu shartli operatorga bog'lik bo'lmagan holda bajariladi.

Agar shart bajarilganda bir nechta operatorlarni bajarish kerak bo'lsa ular figurali qavs (operatorlar qavsi)ga olinadi:

```
if(x < 0) {
```

```
    x = -x;
```

```
    cout<<"Xning qiymati teskari ishorali qiymat bilan  
    almashtiriladi";
```

```
}
```

```
abs = x;
```

Shartli operatorni bir nechta shartlarni bajarilishini tekshirish uchun ham ishlatish mumkin:

```
if(x<0)
```

```
    cout<<"Manfiy qiymat";
```

```
else if(x > 0)
```

```
    cout<<"Musbat qiymat";
```

```
else
```

```
    cout<<"Nol";
```

Tanlash operatori

Tarmoqlanish juda ko'p bo'lganda **if** operatoridan foydalanish qiyinlashadi. Bunday hollarda **switch** – tanlash operatori ishlatiladi.

Tanlash operatorining umumiy ko'rinishi quyidagicha:

```
switch (<ifoda>){
```

```
    case <Nishon1>:<operator1>; break;
```

```
    case <Nishon2>:<operator2>; break;
```

```
    case <NishonN>:<operatorN>; break;
```

```
    default:
```

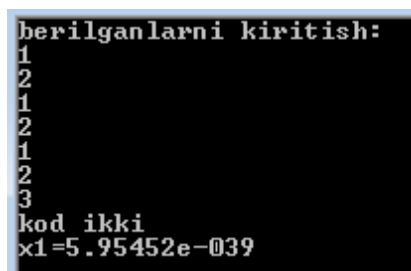
```
        <operator>
```

```
}
```

Xuddi shartli operator kabi tanlash operatorida xam **default** qismini tashlab ketish mumkin.

1-Misol. Masalan, **code** o‘zgaruvchisida 0 dan 2 gacha butun sonlar saqlansin va uning qiymatiga qarab har xil hisoblashlarni amalga oshirish talab qilinsin:

```
#include <iostream>
using namespace std;
#include <math.h>
int main(){
float a,b,c,x,z,y,x1;
int code;
cout<<"berilganlarni kiritish:\n";
cin>>a>>b>>c>>code>>x>>z>>y;
switch (code) {
case 0:
cout << "kod nol"<<"\n"<<"x1="<<x1<<"\n";;
x1 = a*x + 1;
break;
case 1:
cout << "kod bir"<<"\n"<<"x1="<<x1<<"\n";
x1 =sqrt( b*y) + 1;
break;
case 2:
cout << "kod ikki"<<"\n"<<"x1="<<x1<<"\n";
x1 = c*z + 1;
break;
default:
cout << "Ishlanmaydigan qiymat";
}
return 0;
}
```



The screenshot shows a terminal window with the following text:

```
berilganlarni kiritish:
1
2
1
2
1
2
3
kod ikki
x1=5.95452e-039
```

Code o'zgaruvchisining qiymatiga qarab **case** ning biron bir belgisi bajariladi. Operatorning bajarilishi **break** operatori bilan yoki **switch** operatori bilan tugallanadi.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
int s,s1;
float a,b,c,d,x1,x2,x3,z;
cin >> a >> b >> c;
d=b*b-4*a*c; z=2*a;
if (d >0) {
x1=(-b+sqrt(d))/z;
x2=(-b-sqrt(d))/z;
cout<<"x1="<<x1<<"\n";
cout<<"x2="<<x2<<"\n";
}
else if (d =0) {
x1=(-b)/z;
cout<<"x1="<<x1<<"\n"; }
else
cout<<"Ildizi yo'q";
return 0;
}
```

```
5
6
4
Ildizi yo'q
-----
```

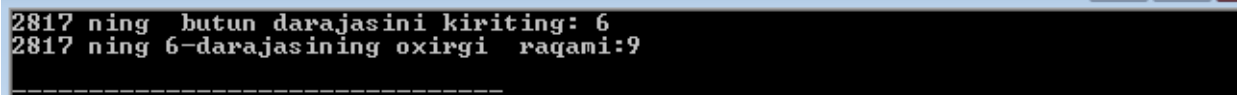
74. 2817^n ning oxirgi raqamini topish.

```
#include<iostream>
using namespace std;
int main(){
int k,n,m;
cout<<"2817 ning butun darajasini kiriting: ";
cin>>n;
k=n%4;
if (k==0){
```

```

    cout<<"2817 ning "<<n<<"-darajasining oxirgi
raqami:"<<1<<endl;
    return 0;}
    if (k==1){
        cout<<"2817 ning "<<n<<"-darajasining oxirgi
raqami:"<<7<<endl;
        return 0;
    }
    if (k==2){
        cout<<"2817 ning "<<n<<"-darajasining oxirgi
raqami:"<<9<<endl;
        return 0;
    }
    if (k==3){
        cout<<"2817 ning "<<n<<"darajasining oxirgi
raqami:"<<3<<endl;
        return 0;
    }
    cout<<"Butun son kiriting!"<<endl;
    return 0;
}

```



```

2817 ning butun darajasini kiriting: 6
2817 ning 6-darajasining oxirgi raqami:9
=====

```

75. $[ax]=b$ tenglamaning dasturini tuzing.

```

#include<iostream>
using namespace std;
int main(){
    float a,b;
    cout<<"a va b ni kiriting:";
    cin>>a>>b;
    if(a!=0){
        cout<<"x="<<a<<" va x<<"<<(b+1)/a<<endl;
        return 0; }
    else
        if (b==0){
            cout<<"x-ixtiyoriy"<<endl;
            return 0; }
}

```

```

else{
cout<<"Tenglama yechimga ega emas"<<endl;
return 0; }
return 0;
}

```

```

a va b ni kiriting:
5
6
x>=5 va x<1.4

```

76. Guruch ikkita xaltaga taqsimlangan. Birinchining og'irligi m kg, ikkinchining og'irligi n kg

- a) qaysi xalta og'irroq, birinchisimi yoki ikkinchisi
- b) og'irroq xaltaning massasini aniqlovchi dasturni tuzing.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int m,n;
cin>>m>>n;
if (m>n)
cout<<"birinchi xalta og'irligi"<< m <<" ga teng";
else cout<<"ikkinchi xalta og'irligi"<<n<<"ga teng";
return 0;
}

```

```

6
5
birinchi xalta og'irligi6 ga teng

```

77. Turistlar o'rmandan A km belgili ustun yaqinidagi yo'lga chiqishdi va yaqindagi avtobus bekatga borishga axt qildi. Xaritaga qarab guruh boshqaruvchisi avtobus bekatlari B km da va C km da joylashganligini aytdi. Turistlar qayerga borsin.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int A,B,C;
cin>>A>>B>>C;
if((A-B)<(C-A))
cout<<B <<"inchi bekatga borish kerak";
else cout<<C <<"inchi bekatga borish kerak";
return 0;
}

```

```
}
```

```
6
5
8
5inchi bekatga borish kerak
```

78. Dildora va Zebo o'zlarining yerlarida A kg qulupnay yig'ishdi, ulardan B kg ni Dildora yig'di. Qizlardan qaysi biri ko'proq qulupnay yig'di va qanchaga ko'p?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int N,A,B;
cin>>A>>B;
N=A-2*B;
if ((A-B)>B)
cout<<"Zebo"<<N<<"ga ko'proq terdi";
else cout<<"Dildora"<<N<<"ga ko'proq terdi";
return 0;
}
```

```
6
4
Dildora-2ga ko'proq terdi
```

79. A m kulrang matoning narxi B so'm, K m ko'k matoning narxi esa M so'm. Qaysi mato qimmatroq va qanchaga.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int A,B,K,M,N;
cin>>A>>B>>K>>M;
if((A*B)>(K*M))
cout<<"kulrang matoning narxi"<<N<<"ga qimmatroq";
else N=K*M-A*B;
cout<<"Ko'k matoning narxi"<<N<<"ga qimmatroq";
return 0;
}
```

```
6
5
7
9
Ko'k matoning narxi33ga qimmatroq
```

80. 1-bo'yoqchilar jamoasi T_1 soatda A m kv devorni, 2-jamoa esa T_2 soatda B m kv devorni bo'yadi. Qaysi jamoaning mexnat ko'rsatkichi balandroq va qanchaga?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
int A,B,T1,T2;
float p;
cin>>A>>B>>T1>>T2;
if((A/T1)>(B/T2)){
p=A/T1-B/T2;
cout<<"birinchi jamoada "<<p<<"ga ko'proq";}
else {
p=B/T2-A/T1;
cout<<"ikkinchi jamoada"<<p<<"ga ko'proq";}
return 0;
}
```

```
6
4
8
6
ikkinchi jamoada0ga ko'proq
```

81. Turist 1 kunda A km ni bosib o'tdi. Tushlikdan oldin u T soat yurib 20 km ni bosib o'tdi. Yana t_2 soat u tushlikdan keyin yurdi. Qachon turistning tezligi balandroq edi. Tushlikdan oldinmi yoki keyinmi?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float A,t1,t2,v1,v2;
cin>>t1>>t2>>A;
v1=20/t1;
v2=(A-20)/t2;
if(v1>v2)
cout<<"turistning tezligi tushlikkacha baland";
else
cout<<"turistning tezligi tushlikdan keyin baland";
return 0;
}
```

```

}
5
6
8
turistning tezligi tushlikkacha baland

```

82. Jism M gr massa va V sm kv hajmga ega. U V_1 sm.kv hajmga va M_1 g massaga ega bo'lgan suyuqlikda suzadimi .

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float M,M1,V,V1,P,P1;
cin>>M>>V>>M1>>V1;
P=M/V;
P1=M1/V1;
if(P<=P1)
cout<<"Jism suzadi";
else
cout<<"Jism suzmaydi";
return 0;
}

```

```

5
6
4
9
Jism suzmaydi

```

83. Iroda va Muborak A kg smarodinadan murobbo qilishmoqchi. Retsept bop'yicha 2 kg smarodinaga 3 kg shakar kerak. Iroda ularga C kg shakar, Muborak esa P kg shakar kerak bo'lishini aytdi. Ulardan qaysi biri haq.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float A,C,P,x;
cout<<"Qancha smorodina yig'ishdi";
cin>>A;
cout<<"Irodaga necha kg shakar kerak edi";
cin>>C;
cout<<"Muborakga necha kg shakar kerak edi";
cin>>P;
if(C=x)

```

```

cout<<"Irodaning gapi to'g'ri";
else if(P=x)
cout<<"Muborakning gapi to'g'ri";
else cout<<"Ikkovlarining gapi noto'g'ri";
return 0;
}

```

```

Qancha smorodina yig'ishdi
5
Irodaga necha kg shakar kerak edi6
Muborakga necha kg shakar kerak edi4
Irodaning gapi to'g'ri

```

84. Sakkiz litr benzinning massasi 5,68 kg. Sisterna 500 m³ hajmga ega. A t benzinni sig'dirish uchun sisterna yetadimi.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float A,x;
cin>>A;
x=8*A/5.68;
if(x<500)
cout<<"Sisterna yetadi";
else
cout<<"Sisterna yetmaydi";
return 0;
}

```

```

10
Sisterna yetadi

```

85. Rux simining 5 m uzunligidagi bo'lagi 430 g massaga ega. Xonadonga simni o'tkazish uchun C m kerak. Bu maqsad uchun M g massali sim o'rami yetadimi?

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float C,x,M;
cin>>C>>M;
x=430*C/5;
if(x<M)
cout<<"Sim o'rami yetadi";
else
cout<<"Sim o'rami yetmaydi";
}

```

```
return 0;
}
```

```
5
6
$in o'rami yetmaydi
```

86. Abdurauf maktabga ketayotib kichik ukasidan 3 minut kech uydan chiqib ketdi. Maktabgacha masofa S m. Abdurauf V_1 m/min tezlik bilan, ukasi esa V_2 m/min tezlik bilan yurishdi. Abdurauf ukasi maktabga yetib borguncha unga yetib oladimi?

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float v1,v2,s,t1,t2;
cin>>v1>>v2>>s;
t1=s/v1;
t2=s/v2;
if((t1-t2)>=3)
cout<<"Abdurauf ukasiga yetib oladi";
else
cout<<"Abdurauf ukasiga yetisholmaydi";
return 0;
}
```

```
6
4
5
Abdurauf ukasiga yetisholmaydi
```

87. Siz kiritgan butun son juft ekanligi haqida fikringizni to'g'ri ekanligini tekshiruvchi dasturni tuzing?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int N;
cout<<"Butun sonni kiriting";
cin>>N;
if(N%2==0)
cout<<"Kiritilgan son juft";
else
cout<<"Kiritilgan son toq";
```

```
return 0;
}
```

```
Butun sonni kiriting
6
Kiritilgan son juft
```

88. Siz kiritgan butun son 3 ga qoldiqsiz bo‘linadi degan fikringizni to‘g‘ri ekanligini tekshiruvchi dasturni tuzing?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int N;
cout<<"Butun sonni kiriting";
cin>>N;
if(N%3==0)
cout<<"Kiritilgan son 3 ga bo‘linadi";
else
cout<<"Kiritilgan son 3 ga bo‘linmaydi";
return 0;
}
```

```
Butun sonni kiriting 63
Kiritilgan son 3 ga bo‘linadi
```

89. Siz kiritgan ikki xonali butun son raqamlarining yig‘indisi juft ekanligini tekshiruvchi dasturni tuzing?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a;
cin>>a;
if((a%10+a/10)%2==0)
cout<<"juft";
else cout<<"juft emas";
return 0; }
```

```
12
juft emas
```

90. Siz kiritgan ikki xonali butun son raqamlarining yig‘indisi 5 ga karrali ekanligini tekshiruvchi dasturni tuzing?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
```

```

int a;
cin>>a;
if((a%10+a/10)%5==0)
cout<<"5 ga karrali";
else
cout<<"5 ga karrali emas";
return 0;
}

```

```

45
5 ga karrali emas

```

91. $y = 5x^2 - 7x + 2$ funksiya grafigi (a,b) koordinatali berilgan nuqtadan o'tishini tekshiruvchi dastur tuzing.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a,b;
cout<<"a va b koordinatalarni kiriting";
cin>>a>>b;
if(b==5*a*a-7*a+2)
cout<<"funksiya grafigi berilgan nuqtadan o'tadi";
else
cout<<"Funksiya grafigi berilgan nuqtadan o'tmaydi";
return 0;
}

```

```

a va b koordinatalarni kiriting
5
4
Funksiya grafigi berilgan nuqtadan o'tmaydi

```

92. Uchta a,b,c haqiqiy son berilgan. Ulardan eng kattasini toping.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a,b,c,max;
cin>>a>>b>>c;
max=a;
if(b>max)
max=b;
if(c>max)

```

```

max=c;
cout<<"3 ta sondan eng kattasi"<<max<<"ga teng";
return 0;
}

```

```

6
5
4
3 ta sondan eng kattasi6ga teng

```

93. Fermer har kuni yog‘liligi K protsentdan kam bo‘lmagan L litr sut beruvchi sigir sotib olmoqchi. Sigir tanlash algoritmi va dasturini tuzing.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int L,K,L1,K1;
cout<<"fermerning rejasi necha litr";
cin>>L;
cout<<"Taklif";
cin>>L1;
cout<<"fermerning yog‘liqlik haqidagi rejasi qanday?";
cin>>K;
cout<<"Taklif";
cin>>K1;
if(L1>=L && K1>=K)
cout<<"Fermer sigirni sotib oladi";
else
cout<<"fermer sigir sotib olmaydi";
return 0;
}

```

```

fermerning rejasi necha litr 25
Taklif 45
fermerning yog‘liqlik haqidagi rejasi qanday? 48
Taklif 26
fermer sigir sotib olmaydi

```

94. Berilgan to‘rtburchak parallelogram ekanligini tekshiruvchi algoritm va dasturini tuzing.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a,b,c,d;
cout<<"to‘rtburchakning tomonlarini kiriting";

```

```

cin>>a>>b>>c>>d;
if (a==c && b==d)
cout<<"to'rtburchak parallelogrammdir";
else
cout<<"to'rtburchak parallelogramm emas";
return 0;
}

```

```

to'rtburchakning tomonlarini kiriting 6
4
5
9
to'rtburchak parallelogramm emas

```

95. Yog'ochli krovat oyoqchasi uchun mo'ljallangan qolip to'rtburchak kesimga ega. Berilgan to'rtburchak shakldagi brusok bu qolipga mos kelishini aniqlang.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float a,b,a1,b1;
cout<<"tirqishning ikkita tomonini kiriting";
cin>>a>>b;
cout<<"Taxtaning ikkita tomonini kiriting";
cin>>a1>>b1;
if(a==a1 && b==b1)
cout<<"Taxta tirqishga to'ri keladi";
else
cout<<"Taxta tirqishga to'g'ri kelmaydi";
return 0;
}

```

```

tirqishning ikkita tomonini kiriting
6
4
Taxtaning ikkita tomonini kiriting
9
8
Taxta tirqishga to'g'ri kelmaydi

```

96. Bitta to'g'ri to'rtburchakning tomonlari A, B ga teng. Boshqasini esa X, Y. To'g'ri to'rtburchaklarning tengligini tekshiruvchi algoritim va dastur tuzing.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){

```

```

float A,X,B,Y;
cout<<"Brinchi to'g'ri to'rtburchakning tomonlarini kiriting";
cin>>A>>B;
cout<<"ikkinchi to'g'ri to'rtburchakning tomonlarini kiriting";
cin>>X>>Y;
if(X==A && Y==B)
cout<<"to'g'ri to'rtburchaklar teng";
else
cout<<"To'g'ri to'rtburchaklar teng emas";
return 0;
}

```

```

Brinchi to'g'ri to'rtburchakning tomonlarini kiriting
6
4
ikkinchi to'g'ri to'rtburchakning tomonlarini kiriting
9
8
To'g'ri to'rtburchaklar teng emas
-----

```

97. Silindrik asbob LO uzunlikka va DO diametrga ega bo'lishi kerak. Yaratilgan asbob me'yorga javob berishini aniqlang.

```

#include <iostream>
using namespace std;
int main(){
float XL,LO,XD,DO;
cout<<"Silindrik detalning uzunligi va diametrini kiriting";
cin>>XL>>XD;
cout<<"detalning uzunlik va diametr me'yorini kiriting";
cin>>LO>>DO;
if(XL==LO && XD==XD)
cout<<"Detal me'yorni qoniqtiradi";
else
cout<<"Detal me'yorni qoniqtirmaydi";
return 0;
}

```

```

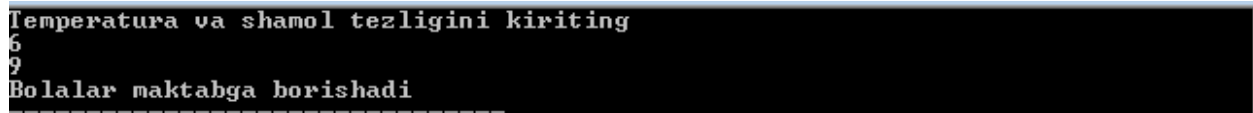
Silindrik detalning uzunligi va diametrini kiriting
6
9
detalning uzunlik va diametr me'yorini kiriting
7
8
Detal me'yorni qoniqtirmaydi
-----

```

98. Boshlang'ich sinflardagi mashg'ulotlar quyidagi hollarda bekor qilinadi, temperatura +25 gradusdan baland bo'lmaganda,

hamda 7 m/s shamoldan baland tezlikda va -20 gradusdan past bo'lmagan temperaturada. Ertalabki ob-havo ma'lumotlaridan bolalar maktabga borishini aniqlang.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int T,S;
    cout<<"Temperatura va shamol tezligini kiriting";
    cin>>T>>S;
    if(T<=-25 || S>7 && T<=-20)
        cout<<"Bolalar maktabga borishmaydi";
    else cout<<"Bolalar maktabga borishadi";
    return 0;
}
```



```
Temperatura va shamol tezligini kiriting
6
9
Bolalar maktabga borishadi
```

99. Bitta uchburchakning tomonlari A1, B1, C1 ga teng. Unga boshqa uchburchakning tengligini tekshiruvchi algoritmi va dasturini tuzing.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int A1,A2,B1,B2,C1,C2;
    cout<<"Birinci uchburchak tomonlarini kiriting";
    cin>>A1>>B1>>C1;
    cout<<"ikkinchi uchburchak tomonlarini kiriting";
    cin>>A2>>B2>>C2;
    if(A1==A2 && B1==B2 && C1==C2)
        cout<<"1-uchburchak ikkinchi uchburchakka teng";
    else
        cout<<"1-uchburchak ikkinchisiga teng emas";
    return 0;
}
```

```

Birinchii uchburchak tomonlarini kiriting
6
9
8
ikkinchi uchburchak tomonlarini kiriting
9
8
7
1-uchburchak ikkinchisiga teng emas

```

100. Odamning normal pulsi minutiga 60 ta urush, bosim 120 ga 80. Kosmanavtlar maktabiga tanlashda puls bo'yicha chetlanish -1 ,+3 bosimning quyi ko'rsatkichi bo'yicha -+3, yuqorisi bo'yicha -+5. Berilgan odamning komissiyadan o'tishini aniqlang.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int P,DN,DV;
cout<<"Odamning pulsini kiriting";
cin>>P;
cout<<"Bosimning yuqori qiymatini kiriting";
cin>>DN;
cout<<"Bosimning quyi qiymatini kiriting";
cin>>DV;
if(P>=59 && DN<=125 && DV>=77)
cout<<"Kiruvchi medkomissiyadan o'tadi";
else cout<<"Kiruvchi medkomissayadan o'tmaydi";
return 0;
}

```

```

Odamning pulsini kiriting
6
Bosimning yuqori qiymatini kiriting
9
Bosimning quyi qiymatini kiriting
8
Kiruvchi medkomissayadan o'tmaydi

```

101. Ikkita bir xil kitob $X < Y < Z$ kattalikni keysga sig'ishini aniqlang (Kitob kattaliklarini o'sib borish tartibida aniqlang).

```

#include <iostream>
using namespace std;
int main(){
float X1,Y1,Z1,X,Y,Z;
cout<<"keys kattaliklarini o'sib borish tartibida kiriting";
cin>>X>>Y>>Z;
cout<<"Kitob kattaliklarini o'sib borish tartibida kiriting";

```

```

cin>>X1>>Y1>>Z1;
if(((2*X1<X)    &&    (2/Y1<Y)    &&    (2/Z1<Z))    ||
((X1<X)&&(2*Y1<Y)&&(Z1<Z))    ||
((X1<X)&&(Y1<Y)&&(2*Z1<Z)))
cout<<"kitoblar keysga sig'adi";
else cout<<"kitoblar keysga sig'maydi";
return 0;
}

```

```

keys kattaliklarini o'sib borish tartibida kiriting
6
9
8
Kitob kattaliklarini o'sib borish tartibida kiriting
4
5
8
kitoblar keysga sig'maydi

```

102. Kompyuterga uchta sportchi uchun suzish bellashuvlarining natijalari kiritilmoqda. Eng yaxshi natijani tanlang va chop eting.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a, b, c;
cout<<"BirinchI sportchining natijasini kiriting a:=";
cin>>a;
cout<<"Ikkinchi sportchining natijasini kiriting b:=";
cin>>b;
cout<<"Uchinchi sportchining natijasini kiriting c:=";
cin>>c;
if((a>b) && (a>c))
cout<<"Eng yahshi natija a:="<<a<<endl;
else if((b>a) && (b>c))
cout<<"Eng yahshi natija b:="<<b<<endl;
else
cout<<"Eng yahshi natija c:="<<c<<endl;
return 0;
}

```

```

BirinchI sportchining natijasini kiriting a:=5
Ikkinchi sportchining natijasini kiriting b:=6
Uchinchi sportchining natijasini kiriting c:=7
Eng yahshi natija c:=7

```

103. Ishchining oylik maoshi ma'lum. Daromad solig'i miqdorini aniqlang.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int MZ,Z;
float N;
cout<<"Minimal oylikni kiriting MZ:=";
cin>>MZ;
cout << "Xodimning oylik moashni kiriting Z:=";
cin>>Z;
if (Z<=5*MZ)
N = Z*0.13;
if ((Z>5*MZ) && (Z<=10*MZ))
N = Z*0.13 + (Z - 5*MZ)*0.20;
else
N = 5 * MZ *0.33 + (Z - 10*MZ)*0.33;
cout<<"Oylik undirilgan soliqlar N:="<<N<<endl;
return 0;
}
```

```
Minimal oylikni kiriting MZ:=1000000
Xodimning oylik moashni kiriting Z:=20000000
Oylik undirilgan soliqlar N:=4950000
=====
```

104. Balli qizlar tanloviga 180 sm dan past bo'lmagan qizlar olinadi. Habibaning tanlovda ishtirok etishga ruxsat berilishini aniqlang.

```
#include<iostream>
using namespace std;
int main(){
int n;
cout<<"Habibaning bo'yini kiriting:";
cin>>n;
if (n>=180)
cout<<"Habiba tanlovda qatnasha oladi"<<endl;
else
cout<<"Habiba tanlovda qatnasha olmaydi"<<endl;
return 0;
}
```

```
}
```

```
Habibaning bo'yini kiriting:158  
Habiba tanlovda qatnasha olmaydi
```

105. Sog'lom kuchukcha kuniga 8 soat o'ynaydi. Tarzan laqabli kuchuk sog'lom ekanligini tekshiruvchi dasturini tuzing.

```
#include <iostream>  
using namespace std;  
int main() {  
    int a;  
    cout<<"Tarzn laqabli kuchikikning kunlik o'ynaydigan vaqtini  
    kiriting a:=";  
    cin >>a;  
    if (a == 8)  
        cout<<"Tarzan laqabli kuchik sog'lom";  
    else  
        cout<<"Tarzan laqabli kuchik kasal";  
    return 0;  
}
```

```
Tarzn laqabli kuchikikning kunlik o'ynaydigan vaqtini kiriting a:=12  
Tarzan laqabli kuchik kasal
```

106. $Y=\sin(X)$ funksiyasi X , $X+1$ oraliqda ildizga ega ekanligini tekshiring. Eslatma; Agar uzluksiz funksiya berilgan oraliqda ildizga ega bo'lsa uning chegaralaridagi qiymat xar hil bo'ladi.

```
#include <iostream>  
#include <math.h>  
using namespace std;  
int main(){  
    int x ;  
    cout<<"Sin(x) argument x nin kiriting x:=";  
    cin >> x;  
    if(sin(x)*sin(x+1)<0)  
        cout<<"y=sin(x) funksiya [x,x+1] oraliqda ildizga ega";  
    else  
        cout<<"y=sin(x) funksiya [x,x+1] oraliqda ildizga ega emas";  
    return 0;}
```

```
Sin(x) argument x nin kiriting x:=1,5  
y=sin(x) funksiya [x,x+1] oraliqda ildizga ega emas
```

107. Yo'lda tezlik 30 km/soatdan oshmasligi kerak. Haydovchi yo'l xarakati qoidalarini buzganligini tekshiring.

```
//Maqsad haydovchimning yo'l harakatini buzganligini
//tekshirish, agar u tezlikni 30km/soatdan oshirsa
//qoidani buzgan hisoblanadi.
#include <iostream>
using namespace std;
int main(){
int b;
cout<<"Haydovchining tezligi b ni kiriting b:=";
cin >> b ;
if (b<30)
cout << " Haydovchi qoidni buzmagani";
else
cout <<" Haydovchi qoidani buzga";
return 0;
}
```

```
Haydovchining tezligi b ni kiriting b:=120
Haydovchi qoidani buzga
```

108. Zanjir qarshiliklarining biri va zanjirning to'la qarshilik qiymatlari bo'yicha bu zanjirda o'tkazgichlar parallel ulash o'rinli ekanligini aniqlans?

```
//bitta zanjirning qarshiligi Rl o'tkazgichning
//qarshiligi R bo'lsa ularni ulab bo'ladimi.
#include <iostream>
using namespace std;
int main(){
float R , Rl;
cout<<"Zanjirning qarshiligini kiriting R:=" ;
cin>>R;
cout<<"Bitta o'tkazgichning qarshiligini kiriting Rl :=" ;
cin>>Rl;
if (R<=Rl)
cout<<"Ularni parallel ulab bo'ladi";
else
cout<<"Ularni parallel ulab bo'lmaydi";
return 0;
}
```

```
}
```

```
Zanjirning qarshiligini kiriting R:=15  
Bitta o'tkazgichning qarshiligini kiriting R1 :=16  
Ularni parallel ulab bo'ladi  
-----
```

109. To'g'ri chiziqqa ma'lum nuqtadan perpendikulyar va og'ma o'tkazilgan. Kesmalar uzunliklari orqali qaysi biri perpendikulyar ekanligini aniqlang?

```
//Maqsad to'g'ri chiziqqa tushurliga perpendikulyar  
//va og'maning qaysi biri perpendikulyar ekanligini  
//aniqlash  
#include <iostream>  
#include <math.h>  
using namespace std;  
int main(){  
int B, I;  
cout<<"Birinch kesmaning uzunligini kiriting B:=";  
cin>>B;  
cout<<"Ikkinchi kesmaning uzunligini kiriting I:=";  
cin>>I;  
if (B>I)  
cout<<"I perpendikulyar";  
else  
cout<<"B perpendikulyar";  
return 0;  
}  
-----  
Birinch kesmaning uzunligini kiriting B:=15  
Ikkinchi kesmaning uzunligini kiriting I:=25  
B perpendikulyar  
-----
```

110. Avtomobilning yaqin masofadagi benzin quyish shaxobchasiga yetgunicha, bakdagi benzin yetadimi?

```
#include <iostream>  
using namespace std;  
int main(){  
float x,y,a,n;  
cout<<"Bir kilometirga sarf qilinadigan benzin miqdorini  
kiriting a =";  
cin>>a;  
cout<<"Yaqin bo'lgan benzin shaxobchasigacha bo'lgan  
masofani kiriting n=";
```

```

cin >>n;
x=a*n;
cout << "Qolgan benzin miqdorini kiriting y=" ;
cin >>y;
if (y>=x)
cout << "Benzin yetadi ";
else
cout <<"Benzin yetmaydi ";
return 0;
}

```

```

Bir kilometirga sarf qilinadigan benzin miqdorini kiriting a =15
Yaqin bo'lgan benzin shaxobchasiigacha bo'lgan masofani kiriting n=12
Qolgan benzin miqdorini kiriting y=6
Benzin yetmaydi

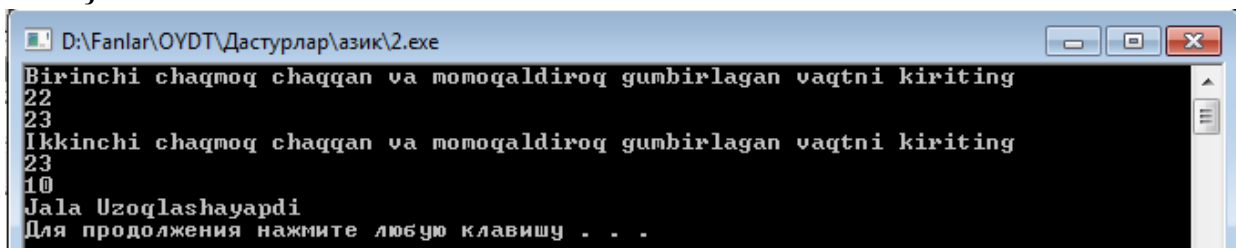
```

111. Chaqmoq chaqishi va momoqalldiroq ovozi oralig'idagi vaqtlarni o'lchab, kelayotgan jala uzoqlashish yoki yaqinlashishini aniqlang.

```

#include <iostream>
using namespace std;
int main(){
float t1, t2, t3,t4;
cout<<"Birinci chaqmoq chaqqan va momoqalldiroq
gumbirlagan vaqtini kiriting "<<endl;
cin >> t1 >> t2;
cout<<"Ikkinchi chaqmoq chaqqan va momoqalldiroq
gumbirlagan vaqtini kiriting "<<endl;
cin >> t3 >> t4;
if((t1-t2)>(t3-t4))
cout<<"Jala yaqinlashayapdi"<<endl;
else
cout<<"Jala Uzoqlashayapdi"<<endl;
return 0;
}

```



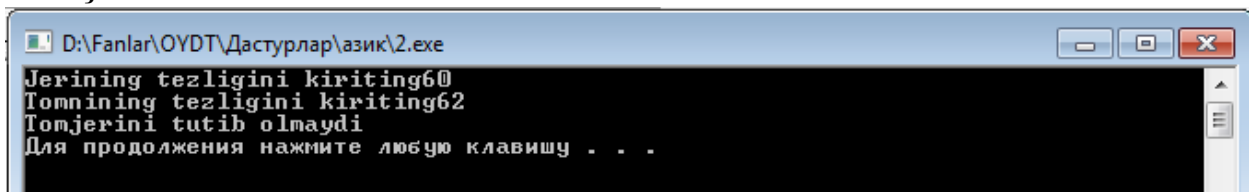
```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Birinci chaqmoq chaqqan va momoqalldiroq gumbirlagan vaqtini kiriting
22
23
Ikkinchi chaqmoq chaqqan va momoqalldiroq gumbirlagan vaqtini kiriting
23
10
Jala Uzoqlashayapdi
Для продолжения нажмите любую клавишу . . .

```

112. Jerri ismli sichqoncha to‘g‘ri chiziq bo‘ylab iniga qochib ketmoqda. Jerri, Tom va inining orasidagi yarim masofada joylashgan bo‘lsa, Tom uni quvib yetadimi?

```
#include <iostream>
using namespace std;
int main(){
float v1,v2;
cout<<"Jerining tezligini kiriting" ;
cin >>v1;
cout<<"Tomnining tezligini kiriting";
cin>>v2;
if(v1>(v2/2))
cout <<"Tomjerini tutib olmaydi"<<endl;
else
cout<<"Tom jeriyni tyutib oldi"<<endl;
return 0;
}
```



113. Har kuni ertalab mayor Xolmatov tirga kirib nishonga 5 ta o‘q uzadi,. Agar u 50 ball to‘plasa , kechqurun teatrqa boradi, kamroq to‘plasa, tirdagi mashg‘ulotlarga boradi. Mayor Xolmatovning kechquringi rejalarini chop etuvchi dasturini tuzing.

```
#include <iostream>
using namespace std;
int main(){
float a1,a2,a3,a4,a5;
cout << "5 ta otishdagi achkolarni kiriting";
cin >> a1 >> a2 >> a3 >> a4 >> a5;
if (a1+a2+a3+a4+a5==50)
cout<<"Mayyor Xolmatov teaterga bordi" << endl;
else
cout<<"Mayyor teaterga bordi";
return 0;
}
```

```

}
5 ta otishdagi achkolarni kiriting5
6
7
6
5
Mayyor teaterga bordiДля продолжения нажмите любую клавишу . . . _

```

114. Butun sonni talab etuvchi va uning 13 dan boshqa barcha qiymatlarini chiqaruvchi dasturini tuzing. Agar qiymati 13 ga teng bo'lsa, uning o'rniga 77 soni chop etilsin.

```

#include <iostream>
using namespace std;
int main(){
int a1, b;
cout<<"a1 ni kiriting" ;
cin >> a1;
if (a1==13) a1=77;
cout <<"Yana ishlaysizmi? 1.Ha/2.Yo'q";
cin >> b;
if (b==1)
cout<<"b="<<b<<endl;
return 0;
}

```

```

a1 ni kiriting12
Yana ishlaysizmi? 1.Ha/2.Yo'q1
b=1

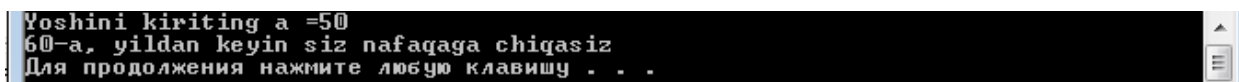
```

115. Odam yoshini so'rovchi va uning nafaqaga chiqqunicha qancha vaqt qolgani yoki uning allaqachon nafaqada ekanligi haqida ma'lumot beruvchi dasturni tuzing.

```

#include <iostream>
using namespace std;
int main(){
int a;
cout << "Yoshini kiriting a =";
cin >> a;
if (a<60)
cout<<"60-a, yildan keyin siz nafaqaga chiqasiz";
else
cout<<"Siz nafaqadasiz";
return 0;
}

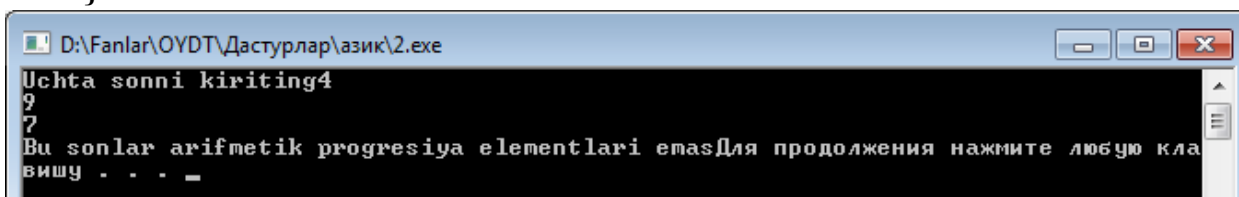
```



```
Yoshini kiriting a =50
60-a, yildan keyin siz nafaqaga chiqasiz
Для продолжения нажмите любую клавишу . . .
```

116. Uchta haqiqiy a , b , c son berilgan. Bu sonlar arifmetik progressiyaning ketma-ket hadlari ekanligi haqida ma'lumot beruvchi dastur tuzing.

```
#include <iostream>
using namespace std;
int main(){
    float a , b , c;
    cout << "Uchta sonni kiriting";
    cin >> a >> b >> c;
    if ((b-a)==(c-b))
        cout<<"Sonlar arifmetik progresiya elementlari";
    else
        cout<<"Bu sonlar arifmetik progresiya elementlari emas";
    return 0;
}
```



```
Uchta sonni kiriting4
9
7
Bu sonlar arifmetik progresiya elementlari emasДля продолжения нажмите любую клавишу . . .
```

117. a , b , c , d – qavariq to'rtburchakning tomon uzunliklar. Unga ichki aylana chizish mumkinligini tekshiruvchi dasturni tuzing.

```
#include <iostream>
using namespace std;
int main(){
    float a , b , c , d ,q;
    cout<<"Qavariq to'rtburchakning tomonlarini kiriting";
    cin>>a>>b>>c>>d;
    if ((a+c) == (b + d))
        cout<<"Qavariq to'rtburchakka ichki aylana chizish mumkin";
    else
        cout<<"Qavariq to'rtburchakka tashqi aylana chizish mumkin
emas";
    return 0;
}
```

```

}
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Qavariq to'rtburchakning tomonlarini kiriting8
9
4
7
Qavariq to'rtburchakka tashqi aylana chizish mumkin emas
Для продолжения нажмите любую клавишу

```

118. Ifodaning qiymati topilsin: $1/x-5$; b) $5/(x-7)(x+3)$.

```

#include <iostream>
using namespace std;
int main(){
float x , y , z;
cout<<"x ning qiymatini kiriting x ="<<endl;
cin>>x;
if(x-5!=0){
y=(1/(x-5));
cout<<"y ="<<y<<endl;
} else
cout<<"x-5==0"<<endl;
if((x - 7!=0)&&(x + 3!=0)){
z=(5/((x - 7)*(x + 3)));
cout<<"z="<<z<<endl; }
else
cout<<" x - 7 = 0 yoki x + 3 = 0";
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
x ning qiymatini kiriting x =
4
y =-1
z=-0.238095
Для продолжения нажмите любую клавишу . . .

```

119. a, b, c koefitsiyentlar berilgan. Bu sonlarning kichigini toping.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(){
float a, b, c, min;
cout<<"koefitsiyentlarni kiriting" <<endl;
cin>>a>>b>>c;

```

```

if (a<b) min=a;
else min=b;
if (min>c) min=c;
cout << "mni =" << min << endl;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
koeffitsiyentlarni kiriting
6
88
9
mni =6
Для продолжения нажмите любую клавишу . . .

```

120. Doira yuzi S_1 va kvadrat yuzi S_2 ma'lum. Kvadrat doiraning ichiga sig'ishini aniqlang.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(){
float const pi = 3.14;
float a, r, S1, S2;
cout<<"Kvadratning yuzini kiriting S1 =";
cin>>S1;
cout<<"Doiraning yuzini kiriting S2 = " ;
cin >> S2;
a= sqrt(S1); r= sqrt(S2/pi);
if ((a*sqrt(r)) <= r)
cout<<"Kvadratga aylana sig'adi"<<endl;
else
cout<<"Kvadratga aylana sig'maydi"<<endl;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Kvadratning yuzini kiriting S1 =36
Doiraning yuzini kiriting S2 = 56
Kvadratga aylana sig'maydi
Для продолжения нажмите любую клавишу . . .

```

121. Doira yuzi S_1 va kvadrat yuzi S_2 ma'lum. Doira kvadratning ichiga sig'ishini aniqlang.

```

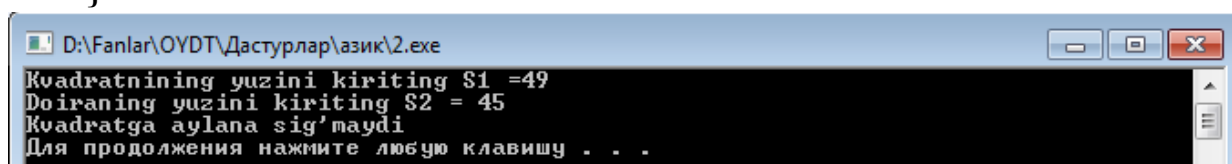
#include <iostream>
#include <math.h>
using namespace std;

```

```

int main(){
float const pi = 3.14; float a, r, S1, S2;
cout << "Kvadratning yuzini kiriting S1 =";
cin >> S1;
cout << "Doiraning yuzini kiriting S2 = " ;
cin >> S2;
a= sqrt(S1); r= sqrt(S2/pi);
if ((a*sqrt(r)) <= r)
cout << "Kvadratga aylana sig'adi" << endl;
else
cout << "Kvadratga aylana sig'maydi" << endl;
return 0;
}

```



```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Kvadratning yuzini kiriting S1 =49
Doiraning yuzini kiriting S2 = 45
Kvadratga aylana sig'maydi
Для продолжения нажмите любую клавишу . . .

```

122. To'g'ri to'rtburchak shakldagi A, B teshikning va g'ishtning X, Y, Z o'lchamlari berilgan. G'isht teshikdan o'ta olishini aniqlang. Eslatma; masalani $A > B$, $X > Y > Z$ deb faraz qilib yeching.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a,b,x,y,z;
cout<<"Teshikning o'lchamlarini kiriting";
cin>>a>>b;
cout<<"G'ishtning o'lchamlarini kiriting";
cin>>x>>y>>z;
if(a>x && b>y || a>y && b>z || a>x && b>z)
cout<<"g'ishtdan o'tadi";
else
cout<<"g'isht teshikdan o'tmaydi";
return 0;
}

```

}

```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Teshikning o'lchamlarini kiriting
7
G'ishtning o'lchamlarini kiriting
3
5
g'ishtdan o'tadi
Для продолжения нажмите любую клавишу . . .
```

123. A, B, C o'lchamli to'g'ri to'rtburchak shakldagi paketni X, Y, Z o'lchamli diplomatga sig'ishini aniqlang.

```
#include <iostream>
using namespace std;
int main(){
float A, B, S, x, y, z;
cout<<"Poketning o'lchamini kiriting" << endl;
cin>>A>>B>>S;
cout<<"Diplomning o'lchamini kiriting " << endl;
cin>>x>>y>>z;
if ((A<=x)&&(B<=y)&&(S<=z))
cout<<"Paketga diplom sig'adi"<<endl;
else
cout<<"Paketga diplom sig'maydi"<<endl;
return 0;
}
```

```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Poketning o'lchamini kiriting
13
15
16
Diplomning o'lchamini kiriting
20
23
25
Paketga diplom sig'adi
Для продолжения нажмите любую клавишу . . .
```

124. Funksiya qiymatini aniqlovchi dastur tuzing.

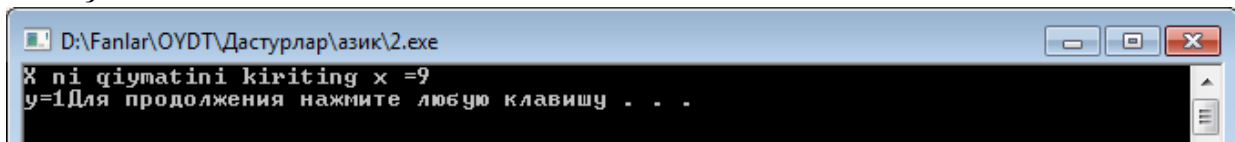
$$y = \begin{cases} x \geq 0 \Rightarrow 1 \\ x < 0 \Rightarrow 0 \end{cases};$$

```
#include <iostream>
using namespace std;
int main(){
int x, y, b;
cout<<"X ni qiymatini kiriting x =";
cin>>x;
```

```

if (x>=0)
y = 1;
else
y = 0;
cout<<"y="<<y;
return 0;
}

```



125. Topishmoq beradigan va javobni tahlil qiladigan dasturni tuzing.

```

#include <iostream>
#include <string.h>
using namespace std;
int main(){
string tj, t, j;
cout<<"Topishmoq javobini kiriting tj =";
cin >> tj;
cout << "Topishmoqni kiriting t =";
cin >> t;
cout << "Topishmoqning javobini ayting j =";
cin >> j;
if(j == tj)
cout<<"Javob to'g'ri"<<endl;
else
cout<<"Javob noto'g'ri"<<endl;
return 0;
}

```



126. To'g'ri burchakli koordinatalar sistemasida 2 ta A (X1, Y1) va B (X2, Y2) nuqtalar koordinatalari berilgan. Bu nuqtadan qaysi biri koordinatalar boshidan uzoqroqda joylashadi.

```

#include <iostream>

```

```

#include <math.h>
using namespace std;
int main(){
float x1, y1, x2, y2, rA, rB;
cout<<"A nuqtaning kordinatasini kiriting"<<endl;
cin >> x1 >> y1 ;
cout<<"B nuqtaning kordinatasini kiriting"<<endl;
cin >> x2 >> y2;
rA = sqrt((x1*x1)+(y1*y1));
rB = sqrt((x2*x2)+(y2*y2));
if (rA > rB)
cout<<"A nuqta B nuqtaga nispatan uzun joylashgan"
<<endl;
else
cout<<"B nuqta A nuqtaga nispatan uzoqroq
joylashgan"<<endl;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
A nuqtaning kordinatasini kiriting
36
67
B nuqtaning kordinatasini kiriting
47
26
A nuqta B nuqtaga nispatan uzun joylashgan
Для продолжения нажмите любую клавишу . . .

```

127. Uchta a, b, c kesma uzunliklari berilgan. Agar ushbu kesmalardan uchburchak yasay olsak, uning perimetri va yuzi xisoblansin.

```

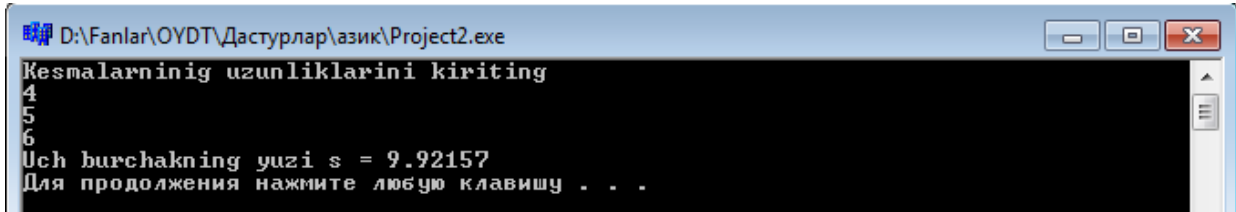
#include <iostream>
#include <math.h>
using namespace std;
int main(){
float a , b ,c , p, pl ,s;
int d;
cout<<"Kesmалarninig uzunliklarini kiriting"<<endl;
cin>>a>>b>>c;
if(((a+b)>c)&&((a+c)>b)&&((b+c)>a))
p = a + b + c;
pl = p /2;

```

```

s = sqrt(pl*(pl - a)*(pl - b)*(pl - c));
cout<<"Uch burchakning yuzi s = "<<s<<endl;
return 0;
}

```



```

D:\Fanlar\OYDT\Дастурлар\азик\Project2.exe
Kesmалarınıg uzunlıklarını kiriting
4
5
6
Uch burchakning yuzi s = 9.92157
Для продолжения нажмите любую клавишу . . .

```

128. Uchta x, y, z son berilgan:

a) $\max(x+y+z, xyz)+3$;

b) $\min(x^2+y^2, y^2+z^2)-4$ topilsin.

```
#include <iostream>
```

```
#include <math.h>
```

```
using namespace std;
```

```
int main(){
```

```
int x, y, z, max, min, a, b;
```

```
cout<<"x, y, z sonlarni kiriting->"<<endl;
```

```
cin >>x>>y>>z;
```

```
if((x+z+y)>(z*x*y))
```

```
max=x+y+z;
```

```
else
```

```
max=z*x*y;
```

```
a=max+3;
```

```
if(((x*x)+(y*y))<((y*y)+(z*z)))
```

```
{
```

```
max=(x*x)+(y*y);}
```

```
else
```

```
{
```

```
max=(y*y)+(z*z);}
```

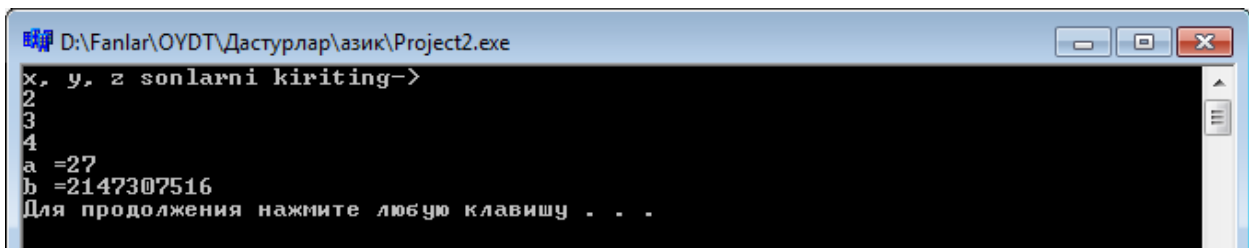
```
b=min-4;
```

```
cout<<"a="<<a<<endl;
```

```
cout<<"b="<<b<<endl;
```

```
return 0;
```

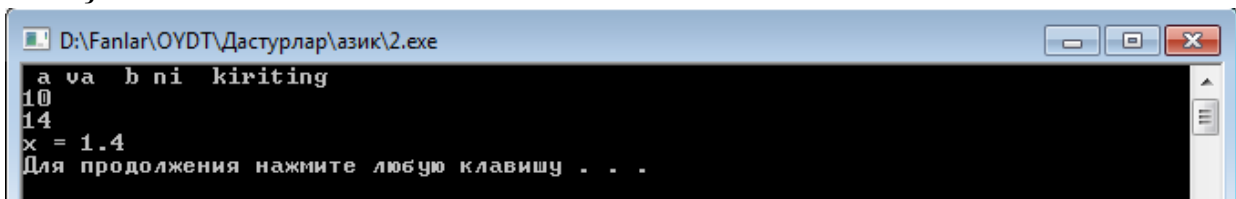
```
}
```



```
x, y, z sonlarni kiriting->
2
3
4
a =27
b =2147307516
Для продолжения нажмите любую клавишу . . .
```

129. $ax=b$ tenglamaning yechimini aniqlovchi dasturni tuzing. Bu yerda a va b berilgan haqiqiy sonlar.

```
#include <iostream>
using namespace std;
int main(){
float x, b, a;
cout << " a va b ni kiriting" << endl;
cin >> a >> b;
if (a!=0)
x =b / a;
cout << "x = " << x << endl;
return 0;
}
```



```
a va b ni kiriting
10
14
x = 1.4
Для продолжения нажмите любую клавишу . . .
```

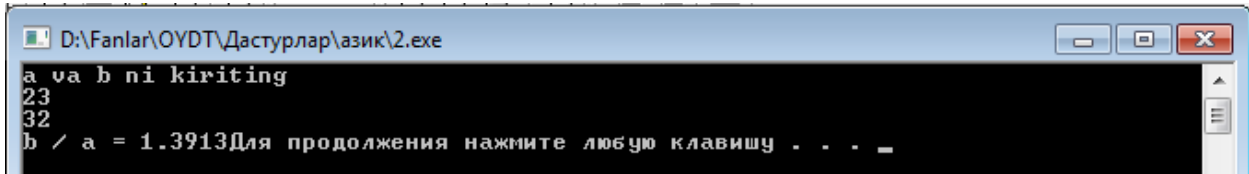
130. Siz kiritgan uchta musbat son orqali bu sonlar uchburchak tomonlari uzunliklari bo‘la olishini tekshiruvchi, va agar bo‘la olsa shu uchburchak turini (o‘tkir burchakli, to‘g‘ri burchakli yoki o‘tmas burchakli) aniqlovchi dasturni tuzing.

```
#include <iostream>
using namespace std;
int main(){
float x, a, b;
cout << "a va b ni kiriting" << endl;
cin >> a >> b;
if(a > 0)
cout << "b / a = " << b/a;
else
if(a < 0 )
cout << "b / a = " << b/a;
```

```

else
if(b > 0)
cout << "Yechimi yo‘q";
return 0;}

```

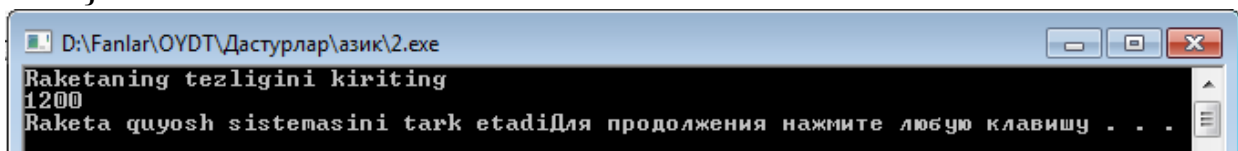


131. Raketa ekvator nuqtasidan o‘chiriladi va tezligini v km/s gacha o‘zgartiradi. O‘chirish natijasi qanday? Eslatma; agar $v \leq 7.8$ km/s bo‘lsa raketa yerga qulab tushadi, agar, $7.8 < v < 11.2$ bo‘lsa raketa yerning yo‘ldoshiga aylanadi, agar $11.2 \leq v \leq 16.4$ bo‘lsa raketa quyoshning yo‘ldoshiga aylanadi, agar $v > 16.4$ bo‘lsa unda raketa quyosh sistemasini tark etadi.

```

#include <iostream>
using namespace std;
int main(){
float v;
cout << "Raketaning tezligini kiriting" << endl;
cin >> v;
if (v <= 7.8 )
cout << "Raketa yerga qulaydi"<< endl;
if ((v >7.8)&&(v < 11.2))
cout << "raketa yerni yo‘ldoshi bo‘ladi";
if ((11.2 <= v)&&(v <= 16.4))
cout <<"Raketa quyoshning yo‘ldosh bo‘ladi";
if (v > 16.4)
cout <<"Raketa quyosh sistemasini tark etadi";
return 0;
}

```



132. Insonning yoshi bo‘yicha tahlil qiluvchi va uni quyidagi to‘rt guruxdan biri ya’ni; maktabgacha, o‘quvchi, ishchi, nafaqachiga kirituvchi dasturni tuzing.

```

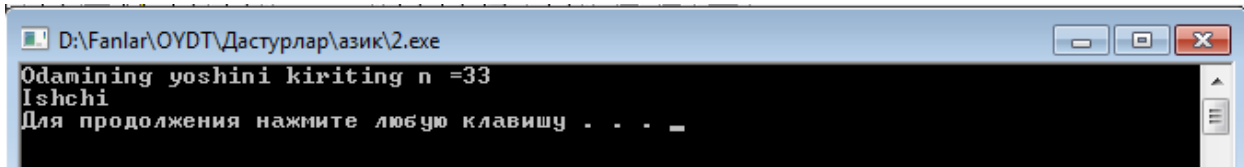
#include <iostream>

```

```

using namespace std;
int main(){
int n;
cout << "Odamining yoshini kiriting n =";
cin >> n;
if (n < 7)
cout << "Maktab yoshigacha bo'lgan bola";
if ((n >= 7) && (n < 17))
cout << "Maktab yoshida"<<endl;
if ((n >= 17)&&(n < 60))
cout << "Ishchi"<< endl;
if (n > 60)
cout << "Nafaqada"<< endl;
return 0;}

```

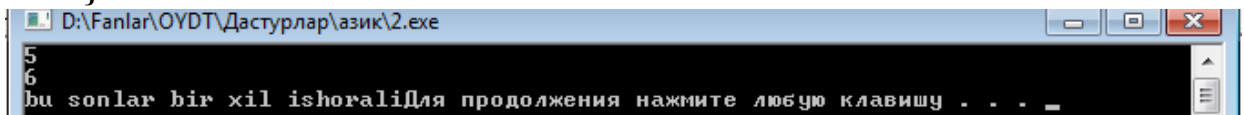


133. a va b sonlar berilgan. Bu sonlar bir xil yoki xar xil ishorali ekanini aniqlang.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a,b;
cin>>a>>b;
if(a*b>0)
cout<<"bu sonlar bir xil ishorali";
else
cout<<"bu sonlar har xil ishorali";
return 0;
}

```



134. Xonadagi temperatura 60° C dan oshganda (uning qanchaligi klaviaturadan kiritiladi) korxonadagi “Yonish holati” ni chiqaruvchi yonish datchig modeli dasturini yozing.

```

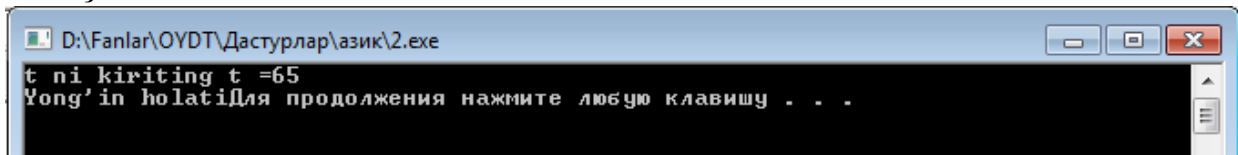
#include <iostream>

```

```

using namespace std;
int main(){
int t;
cout << "t ni kiriting t =" ;
cin >>t;
if (t > 60)
cout << "Yong'in holati";
return 0;
}

```



135. Siz kiritgan raqam kiritilgan uch xonali sonning o'nlik yozuviga kirishini aniqlovchi dasturni tuzing.

```

#include<iostream>
using namespace std;
int main(){
int n,m,k,l,d;
cout<<"Uch xonali son kiriting:";
cin>>n;
k=n%10;
n=(n-k)/10;
l=n%10;
d=(n-l)/10;
cin>>m;
if(m==k||m==l||m==d)
cout<<"kiradi\n";
else cout<<"kirmaydi\n";
return 0;
}

```



136. Ikkita aylana markazlarining (X1,Y1) va (X2, Y2) koordinatalari shuningdek ularning R1 va R2 (R1<R2) radiuslari berilgan. Aylanalarning o'zaro joylashuvini aniqlang.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
L:
int x1, y1, x2, y2, r1, r2;
float r;
cout<<"birinchi aylana koordinatalarini kiriting";
cin>>x1>>y1;
cout<<"Ikkinchi aylana koordinatalarini kiriting";
cin>>x2>>y2;
cout<<"birinchi aylana radiusini kiriting";
cin>>r1;
cout<<"ikkinchi aylana radiusini kiriting";
cin>>r2;
r=sqrt(sqrt(x1-x2)+sqrt(y1-y2));
cout<<"aylanalar kesishmaydi";
goto L;
if(r>r1+r2)
cout<<"aylanalar urinadi";
goto L;
if(r<r1+r2)
cout<<"aylanalar kesishadi";
return 0;
}

```

```

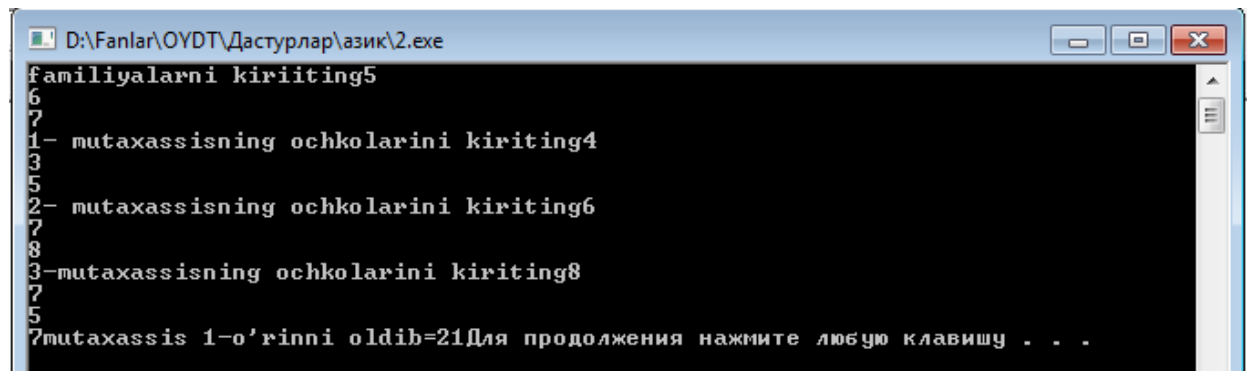
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
birinchi aylana koordinatalarini kiriting10
23
Ikkinchi aylana koordinatalarini kiriting15
17
birinchi aylana radiusini kiriting7
ikkinchi aylana radiusini kiriting8
aylanalar kesishmaydibirinchi aylana koordinatalarini kiriting20
30
Ikkinchi aylana koordinatalarini kiriting19
24
birinchi aylana radiusini kiriting7
ikkinchi aylana radiusini kiriting8
aylanalar kesishmaydibirinchi aylana koordinatalarini kiriting

```

137. “Elektron offis mutaxassisi” kasbi bo‘yicha o‘tkazilgan tanlov yakunida 3 ta ishtirokchi qoldi; Husanov, Shodmonqulov, Sheraliyev. Bellashuvlar uch bosqichda o‘tqazildi. Husanov 1-bosqichda m1, 2-chisida n1, 3-chisida esa p1 ball to‘pladi. Shodmonqulov-mos ravishta m2, n2, p2, Sheraliyev-m3, n3, p3, ball

to‘plashdi. G‘olib necha ball to‘plaganini aniqlovchi va g‘olibning familyasini aniqlovchi dasturni tuzing.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a, a1, a2, a3;
int m1, m2, m3, n1, n2, n3, p1, p2, p3;
int b, b1, b2, b3;
cout<<" ";
cin>>a1>>a2>>a3;
cout<<"1- mutaxassisning ochkolarini kiriting";
cin>>m1>>n1>>p1;
cout<<"2- mutaxassisning ochkolarini kiriting";
cin>>m2>>n2>>p2;
cout<<"3-mutaxassisning ochkolarini kiriting";
cin>>m3>>n3>>p3;
b1=m1+n1+p1;
b2=m2+n2+p2;
b3=m3+n3+p3;
b=b1;
a=a1;
if(b2>b)
b=b2;
a=a2;
if(b3>b)
b=b3;
a=a3;
cout<<a<<"mutaxassis 1-o‘rinni oldi"<<"b="<<b;
return 0;
}
```

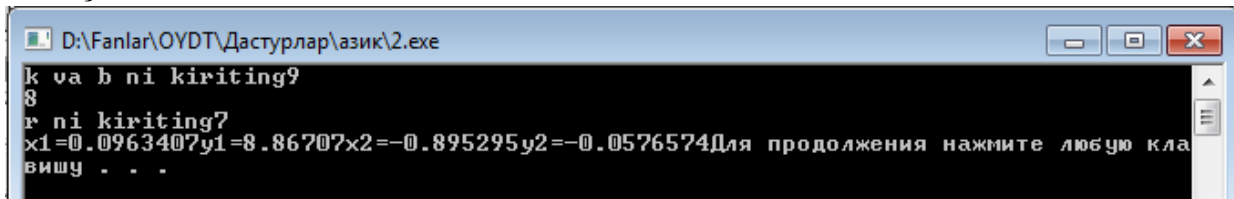


```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
familiyalarni kiriiting5
6
7
1- mutaxassisning ochkolarini kiriting4
3
5
2- mutaxassisning ochkolarini kiriting6
7
8
3- mutaxassisning ochkolarini kiriting8
7
5
7 mutaxassis 1-o'rinni oldib=21Для продолжения нажмите любую клавишу . . .
```

138. $y = kx + b$ to'g'ri chiziq va markazi koordinatalar boshida joylashgan R radiusli aylananing kesishish nuqtalarining koordinatalari topilsin.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
int k, b, r;
float x1, x2, y1, y2;
int m2, n2, p2;
float d;
cout<<"k va b ni kiriting";
cin>>k>>b;
cout<<"r ni kiriting";
cin>>r;
d=4*k*k*b*b-4*(1-k*k)*(b*b-r*r);
if(d>=0){
x1=(-2*k*b+sqrt(d))/(2*(1+k*k));
x2=(-2*k*b-sqrt(b))/(2*(1+k*k));
y2=k*x2+b;
y1=k*x1+b;
cout << "x1=" << x1 << "y1=" << y1 << "x2=" << x2
<< "y2="<<y2;
}
else cout<<"kesishmaydi";
return 0;
```

}



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
k va b ni kiriting?
8
r ni kiriting?
x1=0.0963407 y1=8.86707 x2=-0.895295 y2=-0.0576574
Для продолжения нажмите любую клавишу . . .
```

139. a, b, c koefitsiyentli bikvadrat tenglamaning ildizlarini aniqlovchi dasturni tuzing.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int a, b, c;
    float x1, x2, x3, x4;
    float y1, y2, d;
    cout<<"bikvadrat tenglama koefitsiyentlarini  kiriting";
    cin>>a>>b>>c;
    d=b*b-4*a*c;
    if(d>=0){
        y1=(-b+sqrt(d))/(2*a);
        y2=(-b-sqrt(d))/(2*a);}
    else
        cout<<"tenglama haqiqiy ildizga ega emas";
    if(y1>=0){
        x1=sqrt(y1);
        x2=-sqrt(y1);
        cout<<"x1="<<x1<<"x2="<<x2;}
    else
        if(y2>=0){
            x3=sqrt(y2);
            x4=-sqrt(y2);
            cout<<"x3="<<x3<<"x4="<<x4;}
        else cout<<"Tenglama haqiqiy yechimga ega emas";
    return 0;
```

```

}
D:\Fanlar\OYDT\Дастурлар\азик\Project2.exe
bikvadrat tenglama koefitsiyentlarini kiriting5
6
7
tenglama haqiqiy ildizga ega emasx1=0.000116169x2=-0.000116169Для продолжения на
жмите любую клавишу . . .

```

140. Yilning soni berilgan. Ushbu yildagi kunlar sonini toping. Yo‘riqnoma; Zamonaviy (grigoryan) taqvimda 4 ga bo‘linadigan, (100 ga bo‘linib 400 ga bo‘linmaydiganlarni hisobga olmagan holda, masalan 1900 yilda 29-fevral yo‘q, 2000 yilda esa bor) har yil 29-fevral kuniga ham ega.

```

#include <iostream.h>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int m, n;
cout<<"yil sonini kiriting";
cin>>n;
m=365;
if(n%4==0 && n%400==0)
m=366;
if(n%100==0 && n%400==0)
m=366;
cout<<"m="<<m;
system("PAUSE");
return EXIT_SUCCESS;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
yil sonini kiriting2007
m=365Для продолжения нажмите любую клавишу . . .

```

Mustaqil ishlash uchun vazifa

141. Quyidagi ifodaning qiymatini xisoblaydigan dastur tuzing.

$$a) y = \begin{cases} x > 0 \Rightarrow x^2 \\ x \leq 0 \Rightarrow -x \end{cases}$$

$$b) y = \begin{cases} x \geq 1 \Rightarrow 1 \\ -1 < x < 1 \Rightarrow 0 \\ x \leq -1 \Rightarrow -1 \end{cases}$$

$$c) y = \begin{cases} x > 1 \Rightarrow x - 1 \\ -1 \leq x \leq 1 \Rightarrow 0 \\ x < -1 \Rightarrow x + 1 \end{cases}$$

142. To‘g‘ri burchakli koordinatalar sistemasida 2 ta A (X1, Y1) va B (X2, Y2) nuqtalar koordinatalari berilgan. Nuqtalardan qaysi biri markazi koordinatalar boshida bo‘lgan berilgan radiusli aylanadan uzoqroqda joylashadi?

143. $ax < b$ tengsizlik yechimini aniqlovchi dasturni tuzing. Bu yerda a va b ixtiyoriy haqiqiy sonlar.

144. (x,y) koordinatali nuqta quyidagilardan qaysi biriga tegishli ekanini aniqlang:

a) markazi koordinatalar boshida joylashgan radiusi birga teng bo‘lgan doiraga;

b) markazi koordinatalar boshida joylashgan tashqi radiusi 3 ga va ichki radiusi 2,5 ga teng teshik doiraga.

145. Klaviatura orqali 6 ta sonli tramvay biletining no‘meri kiritiladi. Biletning “baxtli” ekanligini aniqlang.

TAKRORLANUVCHI JARAYONLAR BILAN ISHLASH

Ko‘pincha dasturning qandaydir qismini ko‘p marta takrorlanishiga majbur etish kerak. Takrorlash soni belgilangan, hamda oldindan ma‘lum bo‘lmagan bo‘lishi mumkin. Masalan ma‘lum bir shart bajarilmaguncha dastur qismini takrorlanishiga majbur etishda bunday holat kuzatiladi. Bundan tashqari ma‘lum bir shart bajarilguncha dastur qismini takrorlab turish va sarhat buzilishi bilan takrorlanishlarni bekor qilish ham mumkin.

Takrorlanishlar algoritmik tillarda maxsus operatorlar orqali amalga oshiriladi. Ularni davriy algoritmlar yoki sikllar deb atashadi. Sikllardan foydalanish asosida yaratilgan algoritmlar siklik (davriy) deb ataladi. C++ dasturlash tilida siklik operatorlarning bir necha turlari mavjud.

Avvalo, belgilangan sondagi takrorlanishga ega bo‘lgan siklni ko‘rib chiqaylik. Siklning bunday turi boshqacha qilib aytganda «**for sikli**» deb ataladi. Bu siklda maxsus **siklli o‘zgaruvchi** mavjud bo‘lib, u o‘z qiymatini belgilangan boshlang‘ich qiymatdan belgilangan yakuniy qiymatgacha belgilangan tartibda o‘zgartiradi. Masalan, bu sikl turidan foydalanib belgilangan o‘zgaruvchini ketma-ket ravishda 1, 2, 3,..., 10 (yani 1 dan 10 gacha birlik qadam bilan) qiymatlarni qabul qilishga majburlash mumkin.

Sikl operatori

Tanasidagi operatorlar bir necha bor takrorlanadigan operator takrorlanuvchi (sikl) operatori deyiladi. Uch xildagi sikllar mavjud:

Sharti oldin bajariladigan sikl:

while (shartli_ifoda)

sikl_tanasi;

Sharti keyin tekshiriladigan sikl :

do {

sikl_tanasi;

} **while**(shartli_ifoda);

Iteratsion sikl:

for(siklni_aniklash;shartli_ifoda;ifodalar_ruyxati)

sikl tanasi;

Sikl_tanasi yozish yoki aniqlash uchun foydalanilmaydi. Bunda bo'sh operator, tarkibiy operator, blok bo'lishi mumkin.

Shartli ifoda – bu bajarilish davomiyligini aniqlovchi ifoda.

Sikl aniqlash har doim nuqta vergul bilan tugallanadi.

Siklning tugallanishi quyidagi xollarda ro'y beradi.

- Shartli ifodaning nol qiymatida;
- Sikl tanasidagi bajarishni uning tashqarisiga uzatuvchi operatorlardan (**break**, **goto**, **return**) foydanilganda.

while (shart bajarulguncha tekshir) operatori sharti avval tekshiriladigan operator deyiladi. Siklga kirishda shartli ifoda bajariladi. Agar uning qiymati noldan farqli bo'lsa, u holda **sikl_tanasi** bajariladi. Hisoblash jarayoni shartli ifoda **false** qiymatga ega bo'lmaguncha davom etadi.

while operatori hamma mumkin bo'lgan ketma-ketlikdan foydalanish qulay. Agar uning oxirida qandaydir aniqlovchi belgi bo'lsa. Masalan: quyidagi funksiya qator uzunligini aniqlaydi.

```
int length (char * satr){  
    int len=0;  
    while (satr++) len++;  
    return len; }
```

Bu yerda sikldan chiqish – satr ko'rsatkich adresi ko'rsatilgan qator elementini nolga tenglashtirmoqda. Shartli_ifoda uchun ko'pincha munosabatlardan foydalaniladi. Masalan: natural sonlarning kvadratlarini hisoblaydi.

```
int i=0; //hisoblagich  
int s=0; //Keyingi yig'indi  
while(i<k)  
    s+=i*i; //Yig'indini hisoblash sikli
```

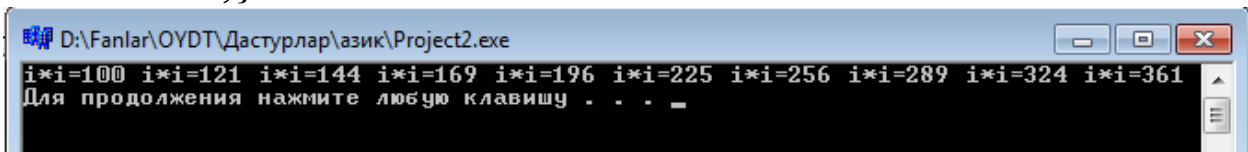
Agar shartli_ifodani nolga tenglashtirish kerak bo'lsa, u holda quyidagi uchta tekshiruv ekvivalentdir:

```
while (point!= NULL) ...  
while (point) ...  
while (point!=0) ...
```

Misollar

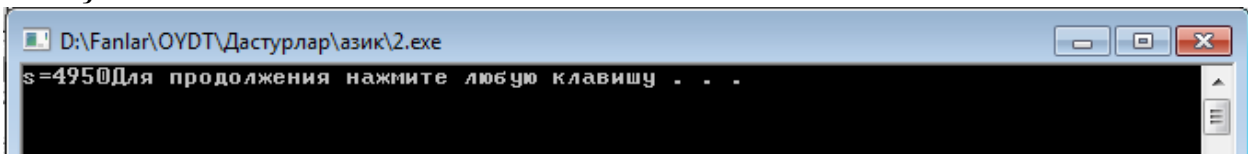
146. 10 dan to 20 gacha bo'lgan sonlarning kvadratlarini ekranga chiqaradigan dasturni tuzing.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int i;
    for(i=10;i<20;i++)
        cout<<"i*i="<<i*i<<"\t";
    return 0;}
```



147. 1 dan to 100 gacha bo'lgan sonlarning yigindisini hisoblovchi dasturni tuzing.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int i,s;
    s=0;
    for(i=1;i<100;i++)
        s=s+i;
    cout<<"s="<<s;
    return 0;
}
```



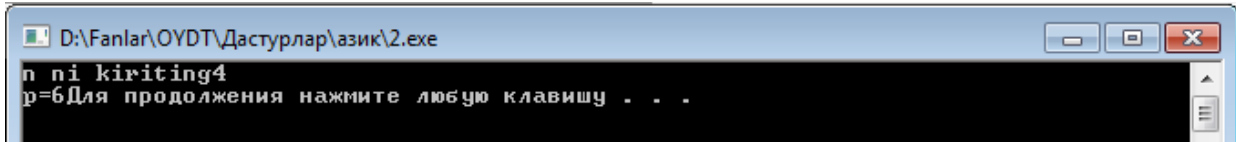
148. 1 dan to N gacha bo'lgan sonlarning ko'paytmasini hisoblovchi dasturni tuzing. N ning qiymati klaviaturadan kiritiladi.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int n,i,p;
    cout<<"n ni kiriting";
```

```

cin>>n;
p=1;
for(i=1;i<n;i++)
p=p*i;
cout<<"p="<<p;
return 0;}

```



149. Klaviatura orqali N ta son kiritiladi. Kiritilgan sonlar ichida manfiy, musbat sonlarning nollar sonining miqdorini aniqlovchi dasturni tuzing. N ning qiymati klaviatura orqali kiritiladi.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int i, n, a[100], k1=0, k2=0, k3=0;
cout<<"n ni kiriting"<<endl;
cin>>n;
cout<<"a ni kiriting"<<endl;
for(i=1;i<=n;i++)
cin>>a[i];
k1=0; k2=0; k3=0;
for(i=1;i<=n;i++){
if(a[i]>0)
k1=k1+1;
else
if(a[i]==0)
k2=k2+1;
else
k3=k3+1; }
cout<<"mus " <<k1<<" man " <<k2<<" nol " <<k3;
return 0;

```

}

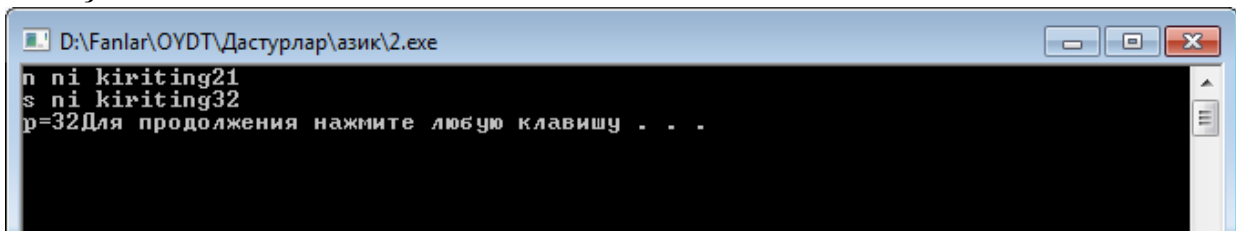
150. Masofani dyumdan santimetrga (1 dyum=2,5sm) o'tkazuvchi jadvalni chop etadigan dasturni tuzing.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int i;
    float a;
    cout<<"Masofani dyumdan sm ga o'tkazadigan jadval";
    for(i=1;i<=20;i++){
        a=i*2.5;
        cout<<i<<" "<<a; }
    return 0; }
```

151. Jamg'arma bankiga yillik 3% daromadli S so'm omonat qo'yildi.N yildan keyin qo'yilma miqdori qancha bo'ladi.

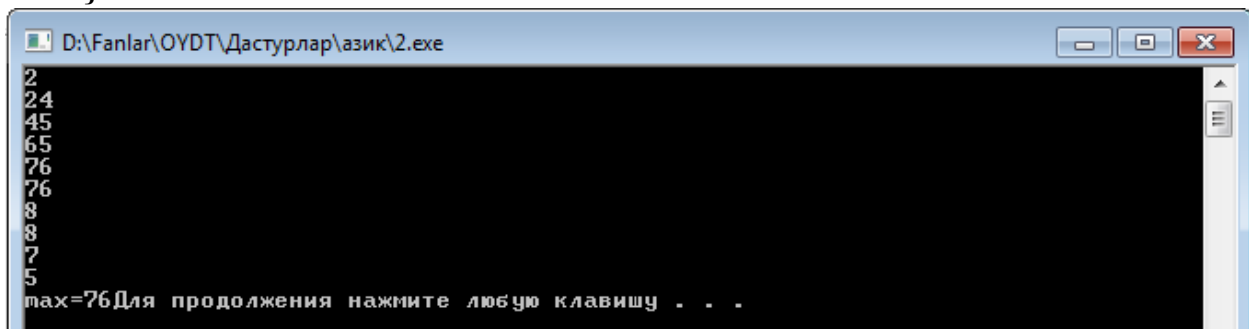
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int s,i,n;
    float p;
    cout<<"n ni kiriting";
    cin>>n;
    cout<<"s ni kiriting";
    cin>>s;
    p=s;
    for(i=1;i<=n;i++)
        p=p*(1+3/100);
    cout<<"p="<<p;
```

```
return 0;
}
```



152. Klaviatura orqali 10 ta son kiritilsin , har bir juftdagi sonlarni taqqoslab, eng kattalari chop etilsin.

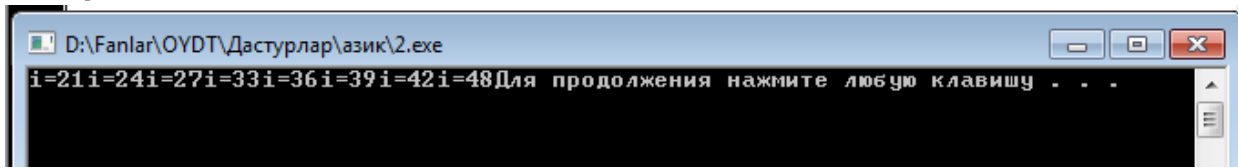
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int i, a[10],max;
for(i=1;i<=10;i++)
cin>>a[i];
max=a[0];
for(i=1;i<=10;i++)
if((i%2==0)&&(max<a[i]))
max=a[i];
cout<<"max="<<max;
return 0;
}
```



153. 20 dan 50 gacha bo‘lgan natural sonlar berilgan. Ulardan 3 ga bo‘linib 5 ga bo‘linmaydiganlari chop etilsin.

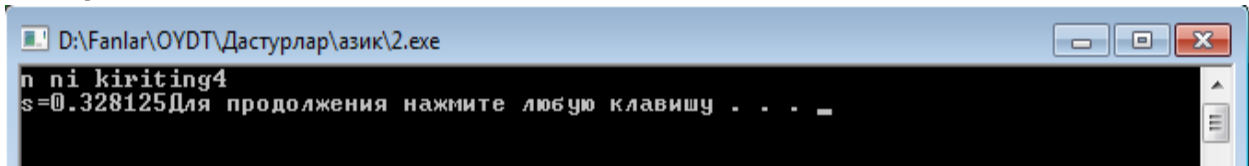
```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int i;
for(i=20;i<=50;i++)
if(i%3==0 && i%5!=0)
cout<<"i="<<i << "\n";
```

```
return 0;
}
```



154. n natural son berilgan. $1/2^2 + 1/4^2 + \dots + 1/(2n)^2$ ni hisoblang.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int i, n;
    float a;
    double s=0;
    cout<<"n ni kiriting";
    cin>>n;
    for(i=1;i<n;i++){
        a=pow(pow(2,i),2);
        s=s+1/a;}
    cout<<"s="<<s;
    return 0;
}
```



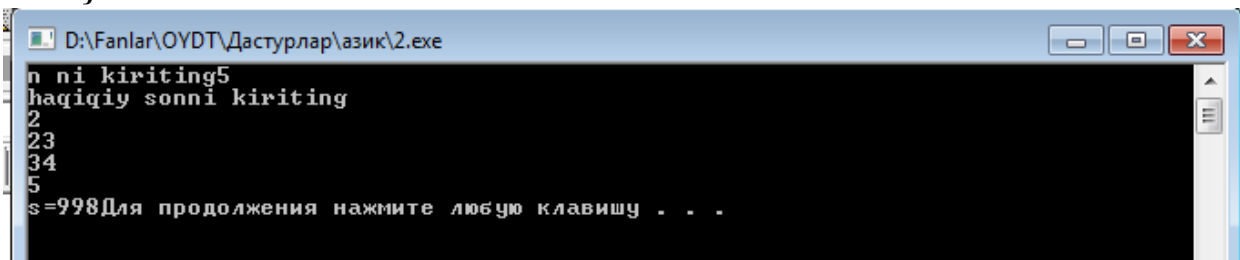
155. n natural $a_1, a_2, \dots, a_n$ haqiqiy sonlar berilgan. $a_1a_2 + a_2a_3 + \dots + a_{n-1}a_n$ ni hosil qiling.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int i, n;
    float a, b, s;
    cout<<"n ni kiriting";
    cin>>n;
    cout<<"haqiqiy sonni kiriting";
    cin>>a;
    b=a;
```

```

s=0;
for(i=2;i<n;i++){
cin>>a;
s=s+a*b;
b=a;
}
cout<<"s="<<s;
return 0;
}

```



```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
n ni kiriting5
haqiqiy sonni kiriting
2
23
34
5
s=998Для продолжения нажмите любую клавишу . . .

```

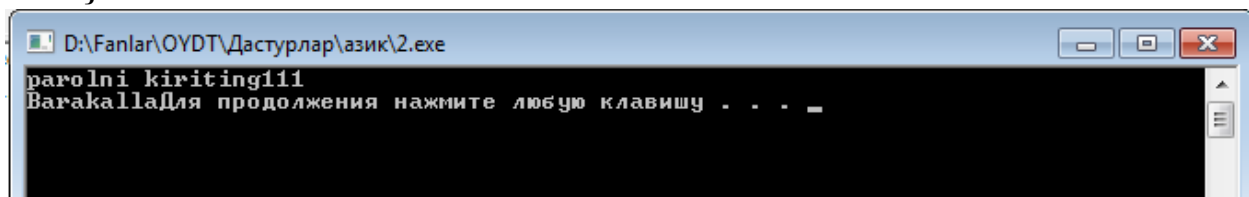
156. “while” operatori yordamida sizdan parol so‘raydigan dastumi tuzing (masalan 111), va agar parol to‘g‘ri bo‘lsa “Barakalla!” xabari yozilsin.

```

#include <iostream>
#include <string.h>
using namespace std;
int main(int argc, char *argv[]){
string passw, answ;
passw="111";
answ="";
while(answ!=passw){
cout<<"parolni kiriting";
cin>>answ;
if(passw!=answ)
cout<<"siz foydalanuvchi emassiz";
else
cout<<"Barakalla";
}
return 0;
}

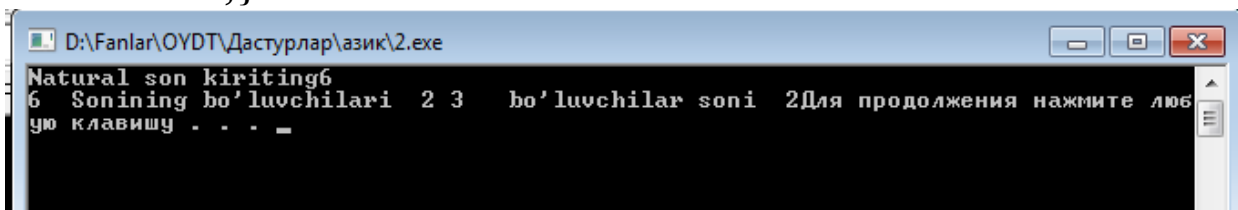
```

}



157. Berilgan sonning bo'luvchilarini o'sish tartibi bo'yicha joylashtiring.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int a, i, nd;
    cout<<"Natural son kiriting";
    cin>>a;
    nd=0;
    cout<<a<<" Sonining bo'luvchilari ";
    for(i=2;i<=a/2;i++)
        if(a%i==0) {
            cout<<i<<" ";
            nd=nd+1;
        }
    cout<<" bo'luvchilar soni "<<nd;
    return 0;}
```



158. Sonni tub ko'paytuvchilarga ajrating.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
    int i, m;
    cout<<"Natural sonni kiriting";
    m10:
    cin>>m;
    if(m<=0)
        goto m77;
```

```

i=1;
m20:
i=i+1;
if(m<=2)
goto m10;
m30:
if((m%i)!=0)
goto m20;
cout<<"i="<<i;
m=m/i;
goto m30;
m77:
return 0;
}

```

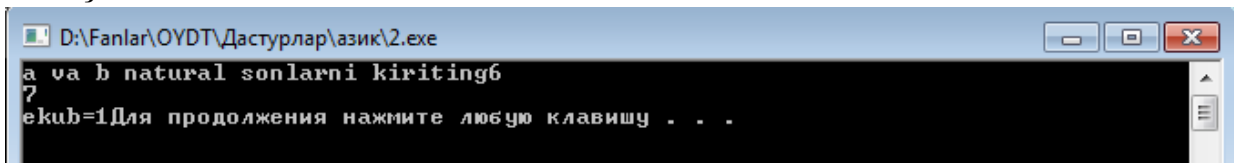
159. Ikkita natural sonning eng katta umumiy bo‘luvchisini aniqlovchi dastur tuzing (Dasturni 2 ta natural sonning eng katta umumiy bo‘luvchisini Evklid algoritmi yordamida topishning quyidagi varianti ko‘rinishida ifodalang: sonlar teng bo‘lmagunicha, kattasi, kattasi va kichigining ayirmasi bilan almashtiriladi. Sonlar bir xil d soniga teng bo‘lib qolganda, ayirish jarayoni to‘xtatiladi: d ikkita sonning eng katta umumiy bo‘luvchisiga teng).

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a, b, r, q, x, y;
cout<<"a va b natural sonlarni kiriting";
cin>>a>>b;
while(a!=b)
if(a>b)
a=a-b;
else b=b-a;
cout<<"ekub="<<a;

```

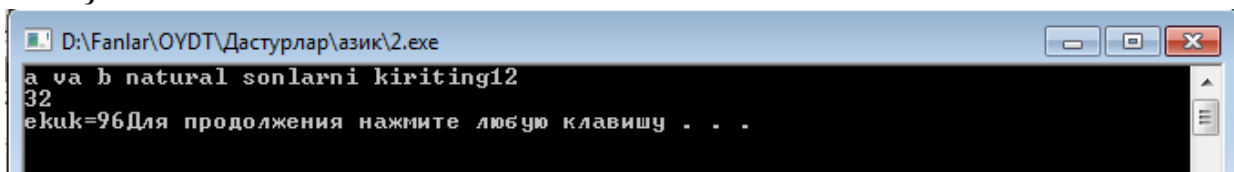
```
return 0;
}
```



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
a va b natural sonlarni kiriting6
7
9
ekub=1Для продолжения нажмите любую клавишу . . .
```

160. Ikkita natural sonlarning eng kichik umumiy karralisini aniqlovchi dasturi tuzing ($EKUK \cdot EKUB = A \cdot B$ formulasidan foydalanilsin).

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int ekuk;
    int a, b, r, q, x, y;
    cout<<"a va b natural sonlarni kiriting";
    cin>>a>>b;
    x=a;
    y=b;
m10:
    q=x/y;
    r=x%y;
    if(r==0)
        goto m20;
    x=y;
    y=r;
    goto m10;
m20:
    ekuk=(a*b)/y;
    cout<<"ekuk="<<ekuk;
    return 0;
}
```



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
a va b natural sonlarni kiriting12
32
96
ekuk=96Для продолжения нажмите любую клавишу . . .
```

161. Agar berilgan $[a, b]$ oraliqda x 0,1 qadam bilan o'zgarsa $y = 3x^2 + x - 4$ funksiyaning eng kichik va eng katta qiymatlari topilsin.

```
#include <iostream>
```

```

#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int a, b;
float x, y, maxy, miny;
cout<<"[a,b] oraliq qiymatlarini kiriting";
cin>>a>>b;
x=a;
y=3*sqrt(x)+x-4;
maxy=y;
miny=y;
while(x<b){
if(y>maxy)maxy=y;
if(y<miny)miny=y;
x=x+0.1;
y=3*sqrt(x)+x-4;}
cout<<"maxy="<<maxy;
cout<<"miny="<<miny;
return 0;
}

```



```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
[a,b] oraliq qiymatlarini kiriting3
2
maxy=4.19615miny=4.19615Для продолжения нажмите любую клавишу . . .

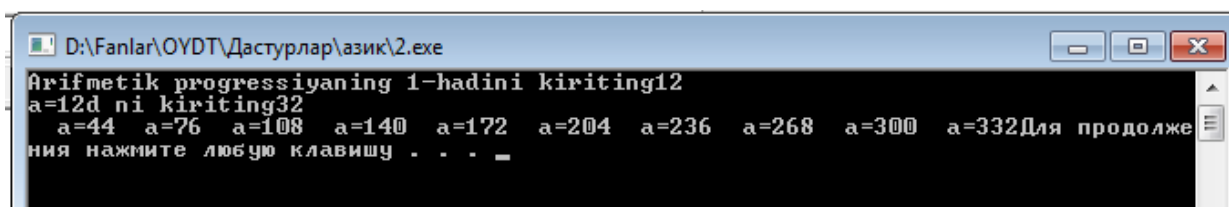
```

162. Arifmetik progressiyaning 10 ta hadi topilsin.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int a, d;
cout<<"Arifmetik progressiyaning 1-hadini kiriting";
cin>>a;
cout<<"a="<<a; cout<<"d ni kiriting";
cin>>d;
for(int i=1;i<=10;i++){
a=a+d;
cout<<" a="<<a;}
return 0; }

```



Mustaqil ishlash uchun vazifa

163. n, m ($n < m$) natural sonlari berilgan. $n, n+1, \dots, m$ sonlari orasida 29-fevral kuniga ega bo'lgan yillar raqamiga teng sonlar nechta ekanligini aniqlang.

164. 1 dan to N gacha bo'lgan sonlarning yig'indisini hisoblovchi dasturni tuzing. N ning qiymati klaviaturadan kiritiladi.

165. Qadimiy uzunlik o'lchovlarini, sotuv va dorixona o'lchovlarini (sikl hisobi 1 dan 10 gacha o'zgaradi) zamonaviyga o'tkazuvchi algoritm va dasturni tuzing.

- a) Sajen metr ga (1 sajen 2,1366 m ga teng)
- b) Fut metr ga (1 fut 0,3048 m ga teng)
- c) Draxm gram ga (1 draxma=3,7325 g)
- d) Unsiy gram ga (1 unsiy=29,86 g)
- e) Funt kilogram ga (1 funt=0,40951 kg)
- f) Arshin metr ga (1 arshin=0,7112 m)
- g) Zolontik gramm ga (1 zolontik=4,2657 g)
- h) Dyum millimetr ga (1 dyum=25,3995 mm)

166. 35 dan 87 gacha bo'lgan natural sonlar berilgan. Ulardan 7 ga bo'linganda 1, 2 yoki 5 qoldiqni beradiganlari topilsin va chop etolsin.

167. 1 dan 50 gacha bo'lgan natural sonlar berilgan. Ulardan 5 ga yoki 7 ga bo'linadigan sonlarning yig'indisini toping.

168. Klaviaturadan 10 ta son kiriting. Agar ular orasida 15 dan katta sonlar bo'lsa ularni 15 ga almashtiring. Hosil bo'lgan sonlarning barchasi chop etilsin.

169. 10 ta musbat va manfiy sonlar kiritilsin. Barcha manfiy sonlar ularning modullari bilan almashtirilsin va hosil bo'lgan 10 ta sonning hammasi chop etilsin.

170. 2 xonali sonlardan 4 ga bo'linib, 6 ga bo'linmaydiganlari chop etilsin.

171. 13 ga karrali, 2 xonali toq sonlarning ayirmasi topilsin.

172. 17 ga karrali 100 dan 200 gacha bo'lgan sonlar yig'indisi topilsin.

173. Klaviatura orqali 10 ta son kiriting. Agar sonning kvadrati 100 dan kichik bo'lsa, sonning o'zi va uning kvadrati chop etilsin.

174. 1 dan to siz kiritgan n butun songacha bo'lgan sonlar kvadratining yig'indisini hisoblovchi dasturni tuzing.

175. a va b sonlarning berilgan qiymatlari orqali a^b ni topuvchi dasturni yozing. E'londa bu o'zgaruvchilarning shartlari kiritilsin (masalan a -kasr bo'lsa, b -manfiy bo'la olmaydi)

176. O'tinni yig'ishtirishda ishtirok etayotgan jamoada N ta o'tin tashish vositasi mavjud. Birinchi vosita m soat har keyingisi esa oldingisiga qaraganda 10 minut ko'proq ishladi. Butun jamoa necha soat ishladi?

177. Kompyuterga N ta nuqtaning koordinatalari kiritiladi. Ulardan nechitasi markazi (a,b) nuqtada bo'lgan R radiusli aylanaga tushishini aniqlang.

178. Kompyuterga sinfdagi N ta o'quvchining bo'ylari haqidagi ma'lumot ketma-ket kiritilmoqda. Sinfdagi o'quvchilarning o'rtacha bo'yini aniqlang.

179. Oddiy kasrlarni qisqartiruvchi dasturni tuzing.

180. N natural son berilgan. N dan katta bo'lmagan va 2, 3, 5 sonlarining hech biriga bo'linmaydigan natural sonlarning miqdorini aniqlang.

181. 2 ta 2 xonali sonlar birin-ketin yozilib, ularning ayirmasiga bo'linadigan 4 xonali sonni tashkil etadilar. Shu sonlar topilsin.

182. 2 ta A va B ikki xonali sonlar berilgan. Bu 2 ta sondan 2 ta to'rt xonali son hosil qilishibdi; birinchi 4 xonali son oldin A sonni keyin B sonni yozishdan hosil bo'libdi; 2 chisi esa oldin B sonni ketin A sonni yozishdan hosil bo'libdi. Agar birinchi to'rt xonali son qoldiqsiz 99 ga 2-chisi esa 49 ga bo'linishi ma'lum bo'lsa, A va B sonlar topilsin.

183. n natural son berilgan. $1/1^1 + 1/2^2 + \dots + 1/n^n$ ni hisoblang.

184. Haqiqiy a va natural n sonlari berilgan. Quyidagilarni hisoblang.

a) $a!$

b) $1 + 1/2 + 1/3 + 1/4 + \dots + 1/n$;

- c) $1+2+4+8+\dots+2^{10}$;
- d) $(1+2)*\dots*(1+2+\dots+10)$;
- e) $a(a+1)\dots(a+n-1)$;
- f) $1/a^2+1/a^2+\dots+1/a^{2n}$;
- g) $\sin x+\sin^2 x+\sin^n x$;
- h) $\sin x+\sin x^2+\dots+\sin x^n$;

185. Barcha 2 xonali sonlarning yig'indisini hisoblovchi dastur tuzing.

186. n natural son berilgan $i=1, 2, \dots, n$ da $b_i=i!$ ekanligini hisobga olib $b_1, b_2, \dots, b_n$ ketma-ketlikni hosil qiling.

187. Birinchi sinf o'quvchisi Azizada m so'm bor. Muzqaymoq k so'm turadi. Aziza muzqaymoqni to'yib yemoqchi bo'ldi, buning uchun u puli yetguncha bittadan muzqaymoq olardi. Aziza oxir-oqibat qancha pul qolishini qanday qilib biladi? Azizaning bo'lishni bilmasligi, u endigina ayirishni va qo'shishni o'rganganligini esdan chiqarmang. U qancha muzqaymoq iste'mol qila oladi.

188. Tadbirkor 20% ostiga bankdan m ming so'm ssuda oldi. Agar u shu vaqt oralig'ida qarzlarini qaytarmay yursa, necha yildan so'ng uning qarzlari S ming so'mdan oshib ketadi.

189. "while" operatori yordamida 2 dan 100 gacha bo'lgan barcha juft sonlarning yig'indisini hisoblovchi dasturni tuzing.

190. "while" operatori yordamida 1 dan 99 gacha bo'lgan barcha toq sonlarning yig'indisini hisoblovchi dasturni tuzing.

191. "while" operatori yordamida $id.og'=bo'y-100$ formulasidan foydalanib katta odamlar uchun ideal bo'yni aniqlaydigan dasturni tuzing. Bo'y qiymati=250 bo'lsa sikldan chiqilsin.

192. M metr uzunlikka ega bo'lgan mato bor. Undan ketma-ket ravishda turli xil uzunlikdagi qismlar qirgilmogda. Matodan foydalanish to'g'risidagi barcha ma'lumotlar kompyuterga kiritiladi. Kompyuter talab qilinayotgan mato qirqimi mavjud bo'lgandagidan uzun bo'lsa, mato yetishmayapganligi haqida ma'lumot berishi kerak.

193. Muzda uchish bellashshuvlari bo'yicha olingan ballar kompyuterga kiritiladi. Har bir chiqishdan so'ng eng yaxshi natijani ekranga chop etuvchi dasturi tuzing.

194. Zanjirda boshlang'ich kuchlanish 200 voltga teng. Ma'lumki u har sekundda quyidagi usul bilan o'zgaradi: bir sekunddan so'ng u $M\%$ ga kamayadi. yana 1 sekunddan so'ng $N\%$ oshadi, so'ngra yana $M\%$ ga kamayadi, keyin $N\%$ oshadi va h.k. Zanjirdagi kuchlanish necha sekunddan so'ng 220 voltdan oshishni tekshuruvchi dasturni tuzing.

195. Klaviatura orqali kiritilgan 2 ta butun son modullarining o'rta geometrigini hisoblovchi dasturni tuzing. Dastur "**while**" siklni qo'llashi kerak. Sikldan chiqib ketish sharti-999 ga teng bo'lgan son qiymatidir.

196. $y=\sin(i/100)$, $i=1, 2, 3, \dots$ ketma-ketlikda birinchi manfiy sonni aniqlovchi dasturni tuzing.

197. i dan katta A haqiqiy son berilgan $1, 1+1/2, 1+1/2+1/3, \dots$ sonlar orasida birinchi A dan katta bo'ladiganini topilsin va uning 0,000001 aniqlikgacha qiymatini chiqaruvchi dasturni tuzing.

198. Klaviatura orqali sonlar kiritiladi va ketma-ketlik bo'yicha yig'iladi Ularning yig'indisi 100 dan oshganda kiritilgan sonlarning miqdori aniqlansin.

199. Mashg'ulotlar boshida, sportchi birinchi kuni 10 km yugirdilar, xar kuni u me'yorni oldingi kun me'yoriga qaraganda 10% ga oshirdi:

a) 7 kunda u umumiy qancha yo'lni yugurib o'tadi?

b) qaysi kuni sportchi birinchi marta 20 km dan ko'proq masofani bosib o'tadi?

c) qaysi kuni sportchining bosib o'tgan yo'lining yig'indisi birinchi marta 60 km dan oshadi?

200. a sonning n natural ko'rsatkichli darajalarini hisoblovchi dasturni tuzing (Dastur variantlarini turli xil sikl ko'rinishlarida **while**, **do while**, **for** bilan yozing).

201. Xokkey o'yini jarayonida jamoalarning jarima vaqtlarini yig'uvchi dasturni tuzing. Ekranga har qanday o'zgarishlardan so'ng har ikkala jamoaning jarima vaqtlar yig'indisi chiqarilsin. O'yin tugagandan so'ng, yakuniy natija chiqarilsin.

202. Har bir bakteriya bir minut ichida 2 taga ajraladi. Boshlang'ich vaqtda bitta bakteriya mavjud bo'ladi. Siz kiritgan

butun vaqt oralig'iga kelib qancha bakteriya mavjudligini hisoblovchi dasturni tuzing.

203. Siz kiritgan sonlardan eng kattasini aniqlovchi dasturni tuzing (Sonlarni kiritish chegarasi bo'lib kiritilgan 0 soni hisoblansin).

204. $b > 0$ haqiqiy sonni berilgan. $a_1, a_2, \dots$, ketma-ketligi quyidagi qoida asosida qurilgan: $a_1 = 1, a_2 = 2 \cdot a_1 + 1, \dots, a_{i+1} = 2 \cdot a_i + 1$ ($i = 2, 3, \dots$) b dan kichik yoki teng barcha $a_1, a_2, \dots$, larni hosil qilish kerak.

205. Natural n va haqiqiy x sonlari berilgan. Quyidagini hisoblang: $\frac{x^1}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$ larni hosil qilish kerak.

206. $a_1 = 1, a_k = k a_{k-1} + 1/k$ ($k = 1, 2, \dots$) bo'lsin. n natural sonni berilgan. a_n ni hosil qiling.

207. $a_1 + 1, a_k = a_{k-1}/k + k$, ($k = 1, 2, \dots$) bo'lsin. n natural sonni berilgan. a_n ni hosil qiling.

208. p (p) zichlikli moddadan tayyorlangan r radiusli shar suvga tushirilgan. Shar markazidan suv yuzasigacha bo'lgan masofani toping.

Yo'llanma. Dasturni tuzish paytida sharning segment balandligi h bo'lgandagi hajim hisoblash formulasidan foydalaning: $V = \pi h^2(3r - h)/3$, bu yerda r –shar radiusi, h –sharning segment balandligi.

209. Berilgan N soni tub son ekanligini aniqlovchi dasturni tuzing.

210. Berilgan N sonidan katta bo'lmagan barcha tub sonlarni ekranga chiqaruvchi dasturni tuzing.

211. Shunday ikki xonali sonlarni topingki, ularning raqamlari yig'indisi kvadrati 13 ga teng bo'linsin.

212. Shunday uch xonali sonlarni topingki, ularning raqamlar yig'indisi kiritilgan natural songa teng bo'lsin.

213. Joriy yilning ixtiyoriy kuni haftanng qaysi kuni bo'lishini aniqlovchi dasturni tuzing.

214. 1202 yilda Italyalik matematik Leonard Pizanskiy (Fibonachchi) shunday masalani taklif etadi: bir juft quyonlar har oy-2 ta quyon (erkak va urg'ochisini) dunyoga keltiradi. Yilning boshida bir juft bo'lgan quyonlar tug'iladi. Yilning boshida bir juft bo'lgan quyonlar yilning oxiriga kelib qanchaga ko'payishadi?

Masala shartiga binoan, har ikki oyda paydo bo'ladigan quyonlar miqdoriga mos keluvchi sonlar 1, 1, 2, 3, 4, 5, 13, 21, 34... ketma-ketlikni hosil qilishadi. Birinchi N ta Fibonachchi sonlarini topuvchi dasturni tuzing.

ALGEBRAIK VA TRANSCENDENT TENGLAMALARNI YECHISH

Qisqacha nazariy kirish. Amaliy hisoblashda ko‘pincha quyidagi ko‘rinishdagi tenglamalarni yechishga to‘g‘ri keladi:

$$f(x)=0;$$

bu yerda $f(x)$ funksiya aniqlangan va berilgan $a < x < b$ chekli yoki cheksiz oraliqda uzliksizdir.

Agar funksiya ko‘phad ko‘rinishda bo‘lsa, $f(x)=0$ tenglama algebraik deb ataladi, agar $f(x)$ funksiya boshqa elementar (trigonometrik, logarifmik, ko‘rsatkichli va h.k.) funksiyalar kirs, bunday funksiyaning transsendent tenglamalar deb ataymiz.

$f(x)$ funksiyaning 0 ga aylantiruvchi har qanday x^* qiymati, ya‘ni $f(x^*)=0$, $f(x)=0$ tenglamaning ildizi deyiladi.

$f(x)=0$ ko‘rinishdagi tenglamaning aniq ildizini ayrim hollardagina topish mumkin. Bundan tashqari, tenglamada ko‘pincha taqribiy koefitsientlar bo‘ladi, bundan ko‘rinib turibdiki, tenglamaning aniq ildizini topish masalasi o‘z ma‘nosini yo‘qotadi. Shuning uchun $f(x)=0$ ko‘rinishdagi tenglamani sonli yechish metodlari ishlab chiqilgan bo‘lib, ular ushbu tenglamaning yaqinlashtirilgan ildiz qiymatlarini topishga imkon yaratadi.

Bu holatda ikkita masalani yechishga to‘g‘ri keladi.

1) Ildizlarni ajratish, ya‘ni har birida aqalli bitta tenglama ildizi mavjud bo‘lgan ancha kichik sohalarni topish.

2) berilgan aniqlikda ildizlarni hisoblash.

$f(x)=0$ tenglamani haqiqiy ildizlari mavjud bo‘lgan sohalarga ajratganda, agar ma‘lum bir kesma oxirida uzluksiz $f(x)$ funksiya turli xil ishorali qiymat qabul qilsa, ushbu kesmada $f(x)=0$ tenglamaning kamida 1 ta ildizga ega bo‘lishidan foydalanishimiz mumkin.

Ikkinchi masalani yechish uchun ko‘plab usullar mavjud bo‘lib, ulardan biz faqatgina bitta usuli (iteratsiya usuli) ni ko‘rib chiqamiz.

Iteratsiya usuli. $f(x)=0$ tenglamani quyidagi ko‘rinishda ifodalaymiz:

$x = \varphi(x)$ buni biz doimo amalga oshirishimiz mumkin. Masalan $f(x)=0$ tenglamadan x ni ajratib, qolganlarini o‘ng tomonga

o'tkazishimiz mumkin. Yoki $f(x)=0$ ning chap va o'ng qismlarini ixtiyoriy λ konstantaga ko'paytiramiz va o'ng va chap qismlariga x ni qo'shamiz, ya'ni $f(x=0)$ ni quyidagicha ifodalaymiz:

$$x = x + \lambda f(x)$$

Bunda $\varphi(x) = x + \lambda f(x)$.

$[\alpha, \beta]$ kesmada ixtiyoriy nuqta x_0 – nollik yaqinlashishni tanlaymiz va keyingi yaqinlashish sifatida quyidagini йфигд qilamiz.

$$x_1 = \varphi(x_0),$$

keyin

$$x_2 = \varphi(x_1),$$

va umuman x_n -quyidagi formula bo'yicha x_{n-1} dan hosil bo'lsin

$$x_n = \varphi(x_{n-1}).$$

Ushbu bo'yicha $x_n (n=1, 2, 3, \dots)$ sonlarni ketma-ket hisoblash jarayonini **iteratsiya** jarayoni deb ataymiz.

Iteratsiya jarayonini ikkita ketma-ket x_{n-1} va x_n yaqinlashishlar uchun quyidagi tenglik bajarilishi ta'minlanmaguncha davom ettirishi kerak.

$$|x_n - x_{n-1}| \leq \varepsilon$$

Bu yerda $\varepsilon - x^*$ ildizning berilgan chegaraviy absolyut xatoligi hisoblanadi.

Misollar

215. Iteratsiya usuli bilan $[-0,5;0]$ kesmada joylashgan, absolyut xatolik $\varepsilon=10^{-4}$ ga ega bo'lgan quyidagi tenglama ildizini toping:

$$F(x)=\arcsin(2x+1)-x^2=0,$$

Shuningdek, ildizni toppish uchun kerak bo'ladigan iteratsiyalar sonini aniqlang. $F(x)=\arcsin(2x+1)-x^2=0$ tenglamani quyidagi usul bilan $x = \varphi(x)$ ko'rinishga keltiramiz:

$$\arcsin(2x+1)=x^2,$$

$$\sin(\arcsin(2x+1))=\sin x^2,$$

$$2x+1=\sin x^2.$$

Bu tenglama osonlikcha quyidagi ko‘rinishga keltirilishi mumkin:

$$x = \varphi(x)$$

bu yerda

$$\varphi(x) = 0,5(\sin x^2 - 1)$$

Boshlabg‘ich yaqinlashish sifatida $[-0,5;0]$ kesmaning ixtiyoriy nuqtasi olinishi mumkin, misol uchun, $x_0 = -0,4$. tenglama ildizini toppish algoritimi quyidagi amallar ketma-ketligini ifodalaydi:

1. $x_0 = -0,4; \varepsilon = 10^{-4}$ va $n=0$ deylik.
2. x_{n+1} yaqinlashishni quyidagi formula orqali hisoblab topamiz:

$$x_{n+1} = 0,5(\sin x_n^2 - 1)$$

3. Quyidagi ayirmani hisoblaymiz $\delta = x_{n-1} - x_n$ va n kattalikni 1 birlikka oshiramiz.

4. Quyidagi shartni tekshiramiz $|\delta| > \varepsilon$. Agar bu shart bajarilsa, keying x_{n+1} yaqinlashishni formula bo‘yicha hisoblashga qaytamiz. Agar shart bajarilmasa, unda natija deb x_{n+1} kattalikni olamiz va hisoblashni yakunlaymiz. Bunda n ning qiymati bajarilgan interasiyalar soniga teng bo‘ladi.

Dasturni tuzushda x_n indeksli o‘zgaruvchini kiritish kerak emas, chunki natija bo‘lib bitta son (tenglama ildizi) olinadi. Hisoblashning har bir pog‘anasida faqatgina 2 ta qo‘shni yaqinlashishni yodda tutish kerak, shuning uchun x_n yaqinlashishni x orqali, x_{n+1} esa y orqali ifodalaymiz. Har bir iterasiya yakunida $x=y$ deb hisoblaymiz.

216. Agar $(\bar{x}, \bar{y})$ nuqta quyidagi sohaga tegishli ekanligi ma’lum bo‘lsa,

$$R(3,5 \leq x \leq 3,6); \quad 2,2 \leq y < 2,3,$$

Quyidagi tenglamalar sistemasining $\bar{x}$ va $\bar{y}$ ildizlarini hisoblovchi dastur tuzing ($\varepsilon = 10^{-4}$);

$$\begin{cases} 2x^2 - xy - 5x + 1 = 0 \\ x - x^2 + 1,6 = 0 \end{cases}$$

Ildizlarni aniqlash uchun iteratsiya usulining quyidagi ko‘rinishidan foydalanamiz:

1) $x_0 = 3,5$ va $y_0 = 2,2$ olinsin.

2) $x_{n+1} = \sqrt{\frac{x_n(y_n+5)-1}{2}}$ va $y_{n+1} = \sqrt{x_n + 1,6}$ hisoblansin.

3) $|x_{n+1} + x_n| \leq \varepsilon$ va $|y_{n+1} - y_n| \leq \varepsilon$ shartlar tekshirilsin.

Agar ikkala shart bajarilgan bo'lsa, 5 chi bo'limni bajarishga o'tilsin; agar ushbu shartlardan atiga bittasi bajarilmagan bo'lsa ham 4 bo'limni bajarishga o'tilsin.

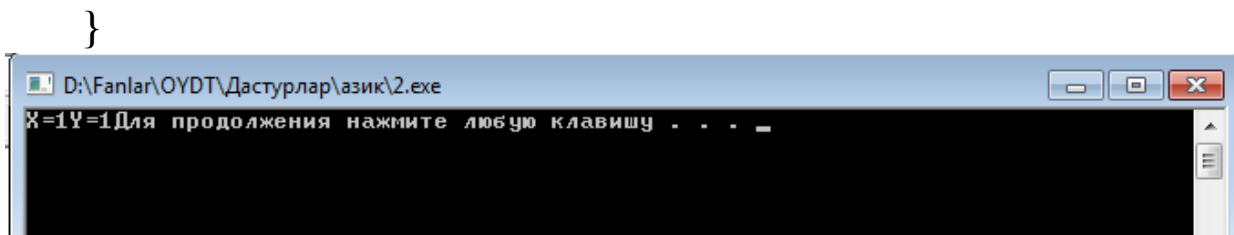
4) $x_n = x_{n-1}, y_n = y_{n+1}$ qo'yilsin va 2 bosqichni bajarishga o'tilsin.

5) Ildizlar qiymati olinsin.

$$\bar{x} = x_{n+1}, \bar{y} = y_{n+1},$$

Quyida tenglamani yechish uchun dastur keltirilgan.

```
#include<iostream>
#include<math.h>
using namespace std;
int main(){
float x, y, eps, delta1, delta2, z1, z2;
int n;
x=3.5;
y=2.2;
eps=0.0001;
n=0;
m2:z1=pow((x*(y+5)-1)/2,1/2);
z2=pow(x+1.6,1/2);
delta1=fabs(z1-x);
delta2=fabs(z2-y);
x=z1;
y=z2;
n++;
if ((delta1>eps)&&(delta2>eps)) goto m2;
cout<<"X="<<x<<"Y="<<y;
return 0;
```



217. Nyuton usuli bilan $[0,5;1]$ kesmada joylashgan, $\varepsilon = 10^{-4}$ absolyut xatolikli quyidagi tenglama ildizi topilsin:

$$f(x) = \sin x - x + 0,15 = 0,$$

$$f'(x) = \cos x - 1 \text{ ni topamiz.}$$

$f(x) = \sin x - x + 0,15 = 0$ tenglama uchun Nyuton usuli bo'yicha hisoblash formulasi quyidagi ko'rinishga ega

$$x_n = x_{n+1} - \frac{\sin x_{n+1} - x_{n+1} + 0,15}{\cos x_{n+1} - 1}$$

bu yerda $n=1,2,3,\dots$

$f(x) = \sin x - x + 0,15 = 0$ tenglama ildizini toppish algoritimi quyidagi amallar ketma-ketligini ifodalaydi:

1. $x_0 = 1, \varepsilon_1 = 0,001$ va $n=0$ ($\varepsilon_1 = 10^{-1} \sqrt{\varepsilon}$) deylik.
2. Quyidagi x_{n+1} , yaqinlashishni quyidgi formula orqali hisoblaymiz.

$$x_{n+1} = x_n - \frac{\sin x_n - x_n + 0,15}{\cos x_n - 1}$$

Quyidagi ayirmaning hisoblaymiz $\delta = x_{n+1} - x_n$ va n kattalikni 1 birlikka oshiramiz.

3. $|\delta| > \varepsilon_1$ shartni tekshiramiz. Agar shart bajarilsa, keyingi x_{n+1} yaqinlashishni $x_{n+1} = x_n - \frac{\sin x_n - x_n + 0,15}{\cos x_n - 1}$ formula bo'yicha

hisoblashga qaraymiz. Agar shart bajarilmasa, natija sifatida x_{n+1} kattalikni olamiz va hisoblashni yakunlaymiz. Bunda n ning qiymati bajarilgan iteratsiya sonlariga teng bo'ladi.

Iteratsiya usulidagidek, indeksli o'zgaruvchini kiritmaymiz, x_n yaqinlashishni esa y orqali ifodalaymiz. Har bir iteratsiya oxirida $x=y$ deb olamiz.

Quyida $x_{n+1} = x_n - \frac{\sin x_n - x_n + 0,15}{\cos x_n - 1}$ tenglamani yechish uchun

dastur keltirilgan.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
float x, y, eps1, delta;
int n;
x=1;
eps1=0.001;
n=0;
m2:
y=x-(sin(x)-x+0.15)/(cos(x)-1);
delta=fabs(y-x);
x=y;
n=n+1;
if(delta>eps1)
goto m2;
cout<<"x="<<x<<"n="<<n;
return 0;
}
```

218. Kesmani 2 ga bo'lish usuli bilan [2;3] kesmada joylashgan $\varepsilon = 10^{-4}$ absolyut xatolikli quyidagi tenglama ildizi aniqlansin.

$$f(x) - x\sqrt{9+x} + x^2 + 4 = 0,$$

tenglama ildizini toppish algoritmi quyidagi amallar ketma-ketligini ifodalaydi:

1. $\alpha = 2$, $\beta = 3$ va $E=0,0001$ deylik ,

$$\alpha - \sqrt{9+\alpha} + \alpha^2 - 4$$

2. $f(\alpha) =$ ni hisoblaymiz.

3. $x=(\alpha + \beta)/2$ ni va shu nuqtadagi $f(x) - x\sqrt{9+x} + x^2 + 4$ qiymatini hisoblaymiz.

4. $f(x)=0$ shartni tekshiramiz. Agar shart $\beta = x$ deb olamiz va hisoblashni yakunlaymiz. Agar shart bajarilmasa, oxirlarida $f(x)$

funksiya har xil ishoraga ega bo'ladigan kesmani tanlashga o'tamiz, aniqroq qilib aytganda:

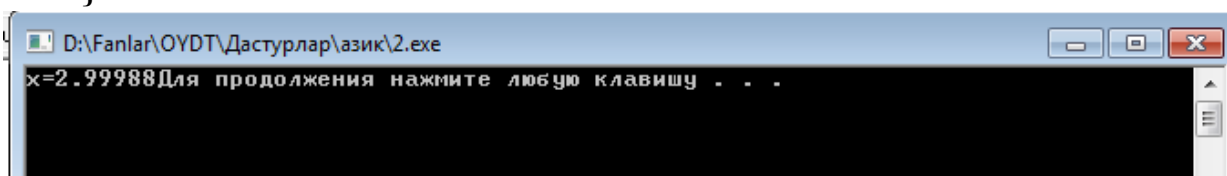
5. $f(\alpha) = f(x)$ shartni tekshiramiz. Agar shart bajarilsa, $\beta = x$ deb olamiz va 6- bo'limga o'tamiz. Agar shart bajarilmasa, $\alpha = x$, $f(\alpha) = f(x)$ deb olamiz va 6- bo'limga o'tamiz.

6. $\beta - \alpha > 2\varepsilon$ shartni tekshiramiz. Agar u bajarilsa, kesmalarni 2 ga bo'lish jarayoniga ya'ni 3-bo'limga qaytamiz. Agar shart bajarilmasa, natija sifatida x ni qabul qilamiz va hisoblashni yakunlaymiz.

Quyida ushbu algoritm asosida tuzilgan dastur keltirilgan.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float a, b, x, eps, fa, fx;
a=2;
b=3;
eps=0.0001;
fa=a-sqrt(9+a)+a*a-4;
m2:
x=(a+b)/2;
fx=x-sqrt(9+x)+x*x-4;
if(fx=0)
goto m7;
if(fa*fx>=0)
goto m5;
b=x;
goto m6;
m5:
a=x;
fa=fx;
m6:
if((b-a)>2*eps)
goto m2;
m7:
cout<<"x="<<x;
```

```
return 0;
}
```



Mustaqil ishlash uchun vazifa

219. Quyidagi jadvalda keltirilgan tenglamalar ildizlarini $\varepsilon = 10^{-4}$ aniqlikda toppish uchun formulalarni yozib oling va yaqinlashish shartini tekshiring. Buning uchun quyidagi ishlarni bazaring:

1. Berilgan kesmadan ildizning boshlang'ich yaqinlashuvini tanlang.
2. Masalani yechadigan dastur tuzing. Dasturda berilgan aniqlikda ildiz qiymatini topishga yordam beruvchi iterasiya sonlarini hisoblash va chop etishga chiqarish hisobga olinsin.
3. Dasturni bajaring, ildiz qiymatini berilgan aniqlikda hosil qiling va uni jadvalda keltirilganlari bilan solishtiring.

	Tenglama	Ildizga ega bo'lgan kesma	Sonli yechish usuli	Ildizning taqribiy qiymati
	$3\sin\sqrt{x} + 0,35x - 3,8 = 0$	[2;3]	Iterasiya	2,2985
	$0,25x^3 + x - 1,2502 = 0$	[0;2]	Nyuton	1,0001
	$x + \sqrt{x} + \sqrt[3]{x} - 2,5 = 0$	[0,4;1]	2ga bo'lish	0,7376
	$x - \frac{1}{3 + \sin 3,6x} = 0$	[0;0,85]	Iterasiya	0,2624
	$0,1x^2 - x \ln x = 0$	[1;2]	Nyuton	1,1183
	$tgx - \frac{1}{3}tg^3x + \frac{1}{5}tg^5x - \frac{1}{3} = 0$	[0;0,8]	2ga bo'lish	0,3333
	$\arccos x - \sqrt{1 - 0,3x^3} = 0$	[0;1]	Iterasiya	0,5629
	$3x - 4\ln x - 5 = 0$	[2;4]	Nyuton	3,2300
	$\cos\frac{2}{x} - 2\sin\frac{1}{x} + \frac{1}{x} = 0$	[1;2]	2ga bo'lish	1,8756

0	$\sqrt{1-0,4x^2} - \arcsin x = 0$	[0;1]	Iterasiya	0,7672
1	$e^x - e^{-x} - 2 = 0$	[0;1]	Nyuton	0,8814
2	$\sin(\ln x) - \cos(\ln x) + 2\ln x = 0$	[1;3]	2ga bo'lish	1,3749
3	$x - 2 + \sin \frac{1}{x} = 0$	[1,2;2]	Iterasiya	1,3077
4	$e^x + \ln x - 10x = 0$	[3;4]	Nyuton	3,5265
5	$\cos x - e^{\frac{x^2}{2}} + x - 1 = 0$	[1;2]	2ga bo'lish	1,0804
6	$1 - x + \sin x - \ln(1+x) = 0$	[0;1,5]	Iterasiya	1,1474
7	$3x - 14 + e^x - e^{-x} = 0$	[1;3]	Nyuton	2,0692
8	$\sqrt{1-x} - \operatorname{tg} x = 0$	[0;1]	2ga bo'lish	0,5768
9	$x + \cos(x^{0,52} + 2) = 0$	[0,5;1]	Iterasiya	0,9892
0	$3\ln^2 x + 6\ln x - 5 = 0$	[1;3]	Nyuton	1,8832
1	$\sin x^2 + \cos x^2 - 10x = 0$	[0;1]	2ga bo'lish	0,1010
2	$x^2 - \ln(1+x) - 3 = 0$	[2;3]	Iterasiya	2,0267
3	$2x\sin x - \cos x = 0$	[0,4;1]	Nyuton	0,6533
4	$e^x + \sqrt{1+e^{2x}} - 2 = 0$	[-1;0]	2ga bo'lish	0,2877
5	$\ln x - x + 1,8 = 0$	[2;3]	Iterasiya	2,8459
6	$x\operatorname{tg} x - \frac{1}{3} = 0$	[0,2;1]	Nyuton	0,5472

7	$tg \frac{x}{2} - ctg \frac{x}{2} + x = 0$	[1;2]	2ga bo'lish	1,0769
8	$0,4 + arctg \sqrt{x} - x = 0$	[1;2]	Iterasiya	1,2388
9	$\sqrt{1-x} - \cos \sqrt{1-x} = 0$	[0;1]	Nyuton	0,4538
0	$0,6 \cdot 3^x - 2,3x - 3 = 0$	[2;3]	2ga bo'lish	2,4200

CHEKLI YIG'INDINI HISOBLASH

Qisqacha nazariy kirish. x ning turli xil qiymatlarida yig'indini $S = \sum_n a_n(x)$ topishga olib kelinuvchi masalalarni ko'rib chiqamiz. Yig'indining har bir qo'shiluvchisi x ning parametriga va ushbu qo'shiluvchining yig'indidagi joyini aniqlovchi n ning nomeriga bog'liq bo'ladi.

Odatda yig'indining umumiy had formulasi quyidagi 3 turdan bittasiga mansub bo'ladi.

$$a) \frac{x^n}{n!}; \quad (-1)^n \frac{x^{2n+1}}{(2n+1)!}; \quad \frac{x^{2n}}{(2n)!};$$

$$b) \frac{\cos nx}{n}; \quad \frac{\sin(2n-1)x}{2n-1}; \quad \frac{\cos 2nx}{4n^2-1}$$

$$c) \frac{x^{4n+1}}{4n+1}; \quad (-1)^n \frac{\cos nx}{n^2}; \quad \frac{n^2+1}{n!} \left(\frac{x}{2}\right)^n.$$

a) holatda yig'indini hisoblash uchun rekurrent munosabatlar, ya'ni keyingi yig'indi hadini oldingisi bilan ifodalash maqsadga muvofiq bo'ladi. Bu hisoblash ish hajmini ancha qisqartiradi. Bundan tashqari yig'indi hadini umumiy formuladan hisoblab topish ko'plab hollarda mumkin emas (masalan $n!$ uchun).

b) holatda rekurrent munosabatlarni qo'llash maqsadga muvofiq emas. Hisoblash jarayoni yig'indining har bir hadi umumiy formula bo'yicha topilganda ancha samarali bo'ladi.

c) holatda yig'indi hadini ikkita ko'paytuvchi sifatida ifodalash maqsadga muvofiq bo'ladi, ulardan bittasi recurrent munosabat bo'yicha, ikkinchisi esa bevosita hisoblab topiladi. Misol uchun,

agar $a_n = \frac{x^{4n+1}}{4n+1}$ bo'lsa, unda $C_n = x^{4n+1}$ deb olinadi va

$C_n = C_{n-1} x^4$ rekurrent ravishda, $\frac{1}{4n+1}$ esa bevosita hisoblab topiladi.

Berilgan qadam bilan ayrim diapazonda o'zgaruvchi yig'indi parametrlari qiymati orqali yig'indini hisoblash masalalarini yechish algoritmi ichma-ich joylashgan ikkita siklda keltiriladi. Ichki sikl x ning fiksirlangan parametrda qo'shiluvchilarni yig'adi, tashqi sikl esa x parametrining o'zgarish qismini tashkillashtiradi. Keyingi jadvalda keltirilgan barcha variantlar uchun hisoblab topiladigan yig'indi ayrim funktsiyanal qator uchun qismli yig'indi hisoblanadi, shuning uchun yig'indini hisoblash bilan bir qatorda $y = f(x)$ funktsiyani hisoblash zarur.

X argumentning o'zgarish diapazoni $a \leq x \leq b$ ko'rinishda berilgan. Yig'indini hisoblashning har bir varianti quyidagi $x=a, a+h, \dots, a+9h, b$ argument qiymatlari uchun keltiriladi, bu yerda

$$h = \frac{b-a}{10}.$$

Misollar

220. Quyidagi yig'indini

$$S = \frac{2}{3} \sin 2x - \frac{3}{8} \sin 3x + \dots + (-1)^N \frac{N}{N^2 - 1} \sin Nx$$

va quyidagi funktsiyani

$$y = \frac{x}{2} \cos x + \frac{1}{4} \sin x$$

$-0.8 \leq x \leq 0.8, N=25$ bo'lgan holda hisoblovchi dastur tuzing.

X ning o'zgarish qadami $h = \frac{0.8 - (-0.8)}{10} = 0.16$ ga teng.

$$S = \frac{2}{3} \sin 2x - \frac{3}{8} \sin 3x + \dots + (-1)^N \frac{N}{N^2 - 1} \sin Nx \quad \text{ni} \quad \text{quyidagi}$$

ko'rinishda yozamiz:

$$S = \sum_{n=2}^{25} (-1)^n \frac{n}{n^2 - 1} \sin nx$$

$$S = \sum_{n=2}^{25} (-1)^n \frac{n}{n^2 - 1} \sin nx \quad \text{ning umumiy hadi quyidagi turga}$$

mansub $c_n = (-1)^n$ ($c_n = -c_{n-1}$) ni rekkurent ravishda hisoblash

maqsadga muvofiq; $\frac{n}{n-1} \sin nx$ ni bevosita hisoblash qulay. Shunday qilib, yig'indi hadi uchun

$$a_n = c_n \frac{n}{n^2 - 1} \sin nx; \quad c_n = -c_{n-1}; \quad n = 2, 3, \dots, 25; \quad c_1 = -1,$$

Yig'indini hisoblashni siklda tashkillashtirish maqsadga muvofiq, bunday siklning har bir o'tishida yig'indining had nomeri 1 ga, yig'indi esa n-chi hadi ya'ni

$$S_n = S_{n-1} + a_n,$$

ga o'zgaradi. Bu yerda $S_n, S_{n-1} - n$ chi hadlar yig'indisi,

$S_n = S_{n-1} + a_n$, formula $n=2, 3, \dots, 25$ holida ko'p marotaba qo'llaniladi.

Dasturni tuzayotganda S_n, c_n indeksli o'zgaruvchilardan foydalanish shart emas, chunki bir paytning o'zida hisoblash jarayonida faqatgina 2 ta qiymat S_n, S_{n-1} va c_n, c_{n-1} ishtirok etadi. Yakuniy natija bo'lib barcha hadlar yig'indisi S_{25} hisoblanadi, shuning uchun bitta, ikkita, uchta va h.k. hadlarning yig'indi qiymatlarini xotirada saqlash shart emas. Bunday holda S_n o'rniga navbatdagi had yig'indisi qo'shilganda o'zgaradigan oddiy S o'zgaruvchidan foydalanish mumkin. Bunda $S = S + c \frac{n}{n^2-1} \sin nx$; formulada S o'ng va chap tomonlarda har xil qiymatga ega: o'ng tomonda oldingi (S_{n+1}) chap tomonda keyingi (S_n) qiymatga ega (c o'zgaruvchi uchun ham xuddi shunday).

Ichki siklning ishi tugagandan keyin S ning qiymati qidirilayotgan yig'indi qiymatiga teng bo'ladi.

Quyida ushbu algoritm asosida tuzilgan dastur keltirilgan.

```
#include <iostream>
```

```
#include <math.h>
```

```
using namespace std;
```

```
int main(){
```

```
float x, h, s, c, y;
```

```
int n;
```

```
x=-0.8;
```

```
h=0.16;
```

```
m5:
```

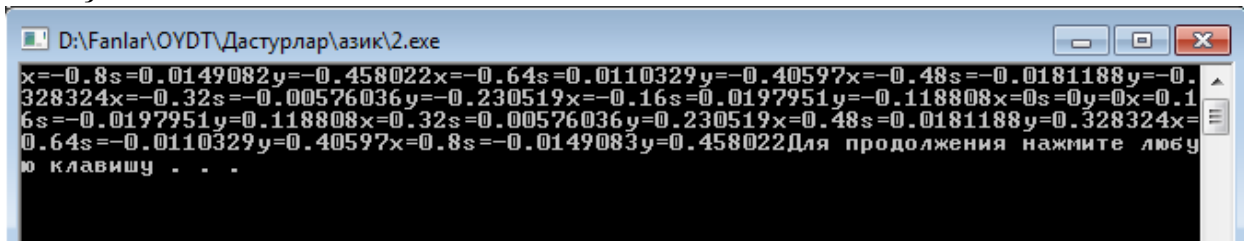
```
s=0;
```

```
c=-1;
```

```

for(n=2;n<50;n++)
c=-c;
s=s+c*n*sin(n*x)/(n*n-1);
y=x*cos(x)/2+sin(x)/4;
cout<<"x="<<x<<"s="<<s<<"y="<<y;
x=x+h;
if(x<=0.8)
goto m5;
return 0;
}

```



221. Quyidagi yig‘indini

$$S = 1 + \frac{x^2}{2!} - \frac{3x^4}{4!} + \dots + (-1)^{N-1} \frac{2N-1}{(2N)!} x^{2N}, N=10 \quad \text{va} \quad y = \cos x + x \sin x$$

funksiyani hisoblovchi dastur tuzing, bunda $0 \leq x \leq 1, h = 0,1$

qadam bilan o‘zgaradi. $S = 1 + \frac{x^2}{2!} - \frac{3x^4}{4!} + \dots + (-1)^{N-1} \frac{2N-1}{(2N)!} x^{2N}, N=10$

ni ushbu ko‘rinishda ifodalaylik:

$$S = \sum_{n=0}^{10} (-1)^{n-1} \frac{2n-1}{(2n)!} x^{2n}.$$

Yig‘indining $a_{n+1} = (-1)^{n-1} \frac{2n-1}{(2n)!} x^{2n}$ umumiy hadi quyidagi turga mansub. $(n+1)$ - chi hadni n -chi orqali ifodalaylik:

$$\begin{aligned}
 a_{n+1} &= (-1)^n \frac{2n-1}{(2(2+n))!} x^{2n+2} \\
 &= -(-1)^{n-1} \frac{2n+1}{(2n)!(2n+1)(2n+2)} x^{2n} x^2 = \\
 &= -a_n \frac{x^2}{(2n-1)(2n+2)}, n = 0, 1, \dots, 9, \quad a_0 = 1.
 \end{aligned}$$

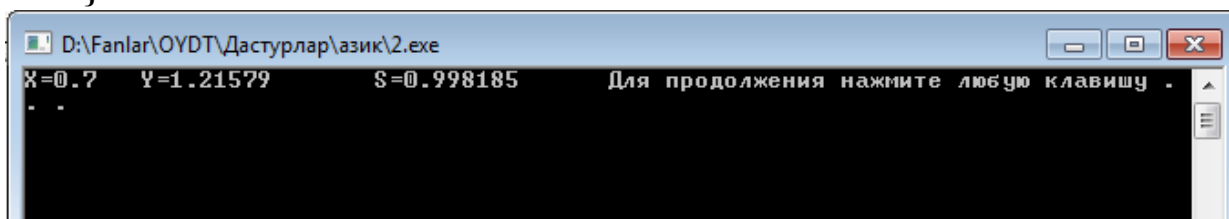
Yig‘indini quyidagi rekkurent formula orqali hisoblaymiz:

$$S_{n+1} = S_n + a_{n+1}, n = 0, 1, \dots, 9; \quad S_0 = 1.$$

220-misolda ifodalangan fikrlar bo'yicha, S_n va a_n indeksli o'zgaruvchilardan foydalanishga ehtiyoj yo'q, shuning uchun dasturda ular o'rniga oddiy S va a o'zgaruvchilar qo'llaniladi.

Quyida ushbu algoritm asosida tuzilgan dastur keltirilgan.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float X,H,S,A,Y;
int k,N,N1;
H=0.1;
for(k=1;k<9;k++)
X=(k-1)*H;
S=1.0;
A=1.0;
for(N=1;N<10;N++)
N1=N-1;
A=-A*X*X/((2*N1-1)*(2*N1+2));
S=S+A;
Y=cos(X)+X*sin(X);
cout<<"X="<<X<<"\t";
cout<<"Y="<<Y<<"\t";
cout<<"S="<<S<<"\t";
return 0;
}
```



Mustaqil ishlash uchun vazifa

222. Qo'shiluvchi va yig'indini hisoblash uchun formulalarni yozib oling. Masalani echadigan dastur tuzing.

	Yig'indi	Argument ning o'zgarish diapazoni		Y funksiyasi
	$S = 1 + \frac{\ln 3}{1!}x + \frac{\ln^2 3}{2!}x^2 + \dots + \frac{\ln^n 3}{n!}x^n$	$0,1 \leq x \leq 1$	10	$y = 3^x$
	$S = \cos x + \frac{\cos 2x}{2} + \dots + \frac{\cos nx}{n}$	$\frac{\pi}{5} \leq x \leq \frac{9\pi}{5}$	40	$y = -\ln \left 2 \sin \frac{x}{2} \right $
	$S = x - \frac{x^3}{3!} + \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!}$	$0,1 \leq x \leq 1$	10	$y = \sin x$
	$S = \sin x - \frac{\sin 2x}{2} + \dots + (-1)^{n+1} \frac{\sin nx}{n}$	$\frac{\pi}{5} \leq x \leq \frac{4\pi}{5}$	40	$y = \frac{x}{2}$
	$S = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$	$1 \leq x \leq 2$	15	$y = e^x$
	$S = 1 + \frac{\cos \frac{\pi}{4}}{1!}x + \dots + \frac{\cos n \frac{\pi}{4}}{n!}$	$0,1 \leq x \leq 1$	25	$y = e^{x \cos \frac{\pi}{4}} \cos(x \sin \frac{\pi}{4})$
	$S = 1 - \frac{x^2}{2!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!}$	$0,1 \leq x \leq 1$	10	$y = \cos x$
	$S = x \sin \frac{\pi}{4} + x^2 \sin 2 \frac{\pi}{4} + \dots + x^n \sin n \frac{\pi}{4}$	$0,1 \leq x \leq 0,8$	40	$y = \frac{x \sin \frac{\pi}{4}}{1 - 2x \cos \frac{\pi}{4} + x^2}$
	$S = x + \frac{x^5}{5} + \dots + \frac{x^{4n+1}}{4n+1}$	$0,1 \leq x \leq 0,8$	30	$y = \frac{1}{4} \ln \frac{1+x}{1-x} + \frac{1}{2} \operatorname{arctg} x$
0	$S = 1 + \frac{\cos x}{1!} + \dots + \frac{\cos nx}{n!}$	$0,1 \leq x \leq 1$	10	$y = e^{\cos x} \cos(\sin x)$
1	$S = 1 + 3x^2 + \dots + \frac{2n+1}{n!}x^{2n}$	$0,1 \leq x \leq 1$	10	$y = (1 + 2x^2)e^{x^2}$
2	$S = \frac{x \cos \frac{\pi}{3}}{1} + \frac{x^2 \cos 2 \frac{\pi}{3}}{2} + \dots + \frac{x^n \cos n \frac{\pi}{3}}{n}$	$0,1 \leq x \leq 0,8$	35	$y = -\frac{1}{2} \ln(1 - 2x \cos \frac{\pi}{3} + x^2)$
3	$S = \frac{x-1}{x+1} + \frac{1}{3} \left(\frac{x-1}{x+1} \right)^3 + \dots + \frac{1}{2n+1} \left(\frac{x-1}{x+1} \right)^{2n+1}$	$0,1 \leq x \leq 1$	10	$y = \frac{1}{2} \ln x$
4	$S = -\cos x + \frac{\cos 2x}{2^2} + \dots + (-1)^n \frac{\cos nx}{n^2}$	$\frac{\pi}{5} \leq x \leq \pi$	20	$y = \frac{1}{4} \left(x^2 - \frac{\pi^2}{3} \right)$
5	$S = \frac{x^3}{3} - \frac{x^5}{15} + \dots + (-1)^{n+1} \frac{x^{2n+1}}{4n^2 - 1}$	$0,1 \leq x \leq 1$	30	$y = \frac{1+x^2}{2} \operatorname{arctg} x - \frac{x}{2}$

6	$S = \sin x + \frac{\sin 3x}{3} + \dots + \frac{\sin(2n-1)x}{2n-1}$	$\frac{\pi}{10} \leq x \leq \frac{9\pi}{10}$	40	$Y = \frac{\pi}{4}$
7	$S = 1 + \frac{x^2}{2!} + \dots + \frac{x^{2n}}{(2n)!}$	$0.1 \leq x \leq 1$	10	$Y = \frac{e^x + e^{-x}}{2}$
8	$S = \frac{\cos 2x}{3} + \frac{\cos 4x}{15} + \dots + \frac{\cos 2nx}{4n^2 - 1}$	$0.1 \leq x \leq 0.8$	50	$Y = \frac{1}{2} - \frac{\pi}{4} \sin x $
9	$S = 1 + 2 \frac{2x}{1!} + \dots + \frac{(2x^n)}{n!}$	$0.1 \leq x \leq 1$	20	$Y = e^{2x}$
0	$S = 1 + 2 \frac{x}{2} + \dots + \frac{n^2 + 1}{n!} \left(\frac{x}{2}\right)^n$	$0.1 \leq x \leq 1$	30	$Y = \left(\frac{x^2}{4} + \frac{x}{2} + 1\right)e$
1	$S = x - \frac{x^3}{3} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1}$	$0.1 \leq x \leq 0.5$	40	$Y = \operatorname{arctg} x$
2	$S = 1 - \frac{3}{2}x^2 + \dots + (-1)^n \frac{2n^{2+1}}{(2n)!} x^{2n}$	$0.1 \leq x \leq 1$	35	$Y = \left(1 - \frac{x^2}{2}\right) \cos x - \frac{x}{2} \sin x$
3	$S = -\frac{(2x)^2}{2} + \frac{(2x)^4}{24} + \dots + \frac{(2x)^{2n}}{(2n)!}$	$0.1 \leq x \leq 1$	15	$Y = 2(\cos^2 x - 1)$
4	$S = -(1+x)^2 + \frac{(1+x)^4}{2} + \dots + (-1)^n \frac{(1+x)^{2n}}{(n)}$	$-2 \leq x \leq -0.1$	40	$Y = \ln \frac{1}{1+2x+x^2}$
5	$S = + \frac{x^3}{3!} + \dots + \frac{x^{2n+1}}{(2n+1)!}$	$0.1 \leq x \leq 1$	20	$Y = \frac{e^x - e^{-x}}{2}$
6	$S = \frac{x}{3!} + \frac{4x^2}{5!} + \dots + \frac{n^2}{(2n+1)!} x^n$	$0.2 \leq x \leq 0.8$	20	$Y = \frac{1}{4} \left(\frac{x+1}{\sqrt{x}} \operatorname{sh} \sqrt{x - ch \sqrt{x}} \right)$
7	$S = x \cos \frac{\pi}{4} + x^2 \cos 2 \frac{\pi}{4} + \dots + x^n \cos n \frac{\pi}{4}$	$0,1 \leq x \leq 0,8$	40	$y = \frac{x \cos \frac{\pi}{4} - x^2}{1 - 2x \cos \frac{\pi}{4} + x^2}$
8	$s = 3x + 8x^2 + \dots + n(n+2)x^n$	$0,1 \leq x \leq 0,8$	40	$y = \frac{x(3-x)}{(1-x)^3}$
9	$s = \cos x \frac{\cos 3x}{3^2} + \dots + \frac{\cos(2n-1)x}{(2n-1)^2}$	$\frac{\pi}{5} \leq x \leq \pi$	40	$y = \frac{\pi^2}{8} - \frac{\pi}{4} x $
0	$s = \frac{x^2}{2} - \frac{x^4}{12} + \dots + (-1)^{n+1} \frac{x^{2n}}{2n(2n-1)}$	$0,1 \leq x \leq 0,8$	10	$y = x \operatorname{arctg} x - \ln \sqrt{1+x^2}$

ODDIY MASSIVLAR BILAN ISHLASH

Massiv ko'rinishidagi ma'lumotlar bir hil turdagi nomerlangan elementlarning fiksirlangan sonini o'zida mujassam etadi. C++ da massivlarni tavsiflashning umumiy ko'rinishi quyidagicha bo'ladi:

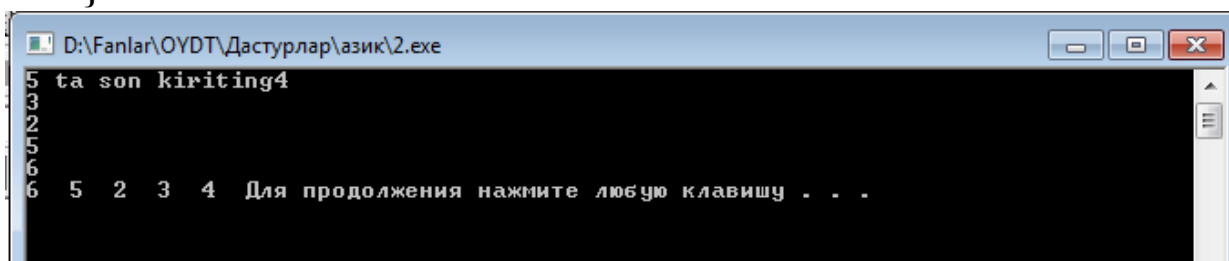
```
<tur> <o'zgaruvchi_nomi>[<elementlar_soni>];
```

By yerda <tur> massiv elementlarining turi; <o'zgaruvchi_nomi> massiv o'zgaruvchisining nomi; kvadrat qavslar ichida <elementlar_soni> massiv nechta elementlardan tashkil topishi ko'rsatiladi.

Misollar

223. Beshta familyadan iborat massiv tuzing va uning elementlarini oxirgisidan boshlab ustun shaklida chiqaring.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int a[5];
    int i;
    cout<<"5 ta son kiriting";
    for (i=1;i<=5;i++)
        cin>>a[i];
    for (i=5;i>=1;i--)
        cout<<a[i]<<" ";
    return 0;
}
```



224. 5 ta sondan iborat massiv tuzing. Klaviaturiadan kiritilgan sondan boshlanuvchi familyani ekranga chiqaring.

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int a[5];
```

```

int i;
int x;
cout<<"5 ta sonidan iborat massiv tuzing";
for(i=0;i<=5;i++)
cin>>a[i];
cout<<"Qaysi sonni chop etmoqchisiz";
cin>>x;
for(i=1;i<=5;i++)
if(x==a[i])
cout<<a[i];
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
5 ta sonidan iborat massiv tuzing23
42
44
55
34
24
Qaysi sonni chop etmoqchisiz
1
Для продолжения нажмите любую клавишу . . .

```

225. A[1..7] massivni tasodifiy sonlar generatori yordamida hosil qiling va uni ekranga chiqaring. Uning barcha elementlarini 2 marta oshiring.

```

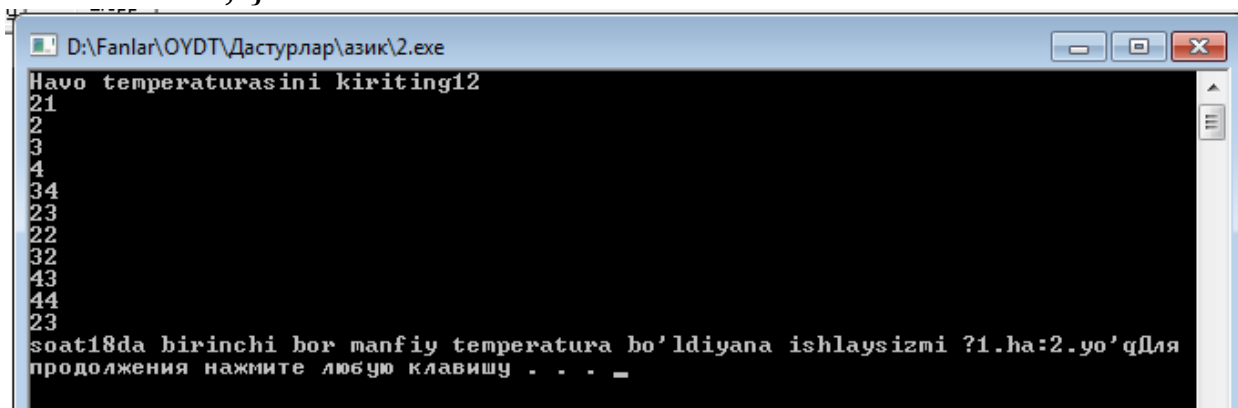
#include <stdio.h>
#include <stdlib.h>
#include <iostream>
#include <time.h>
using namespace std;
int main(int argc, char *argv[]){
int a[7];
int i,b;
srand (time(NULL));
cout<<"Ehtimollik sonlar generatori tuzilmoqda";
for(i=0;i<7;i++){
a[i] = rand() % 20 + 1;
cout<<a[i] << " \t";}
for(i=0;i<7;i++){
a[i]=2*a[i];
cout<<a[i]; }

```

```
return 0; }
```

226. Soat 8 dan 20 gacha havoning temperaturasi har soat o'lganib turiladi. Bu vaqt oralig'ida temperatura pasayganligi ma'lum. Soat nechada birinchi marta manfiy temperatura kuzatiladi?

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
    int t[12];
    int i,clock,b;
    L:
    cout<<"Havo temperaturasini kiriting";
    for(i=1;i<13;i++)
        cin>>t[i];
    clock=7;
    for(i=1;i<12;i++){
        clock=clock+1;
        if(t[i]<0)
            goto L1;}
    L1:
    cout<<"soat"<<clock<<"da birinchi bor manfiy temperatura
bo'ldi";
    cout<<"yana ishlaysizmi ?1.ha:2.yo'q";
    if(b==1)
        goto L;
    return 0; }
```



```
D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Havo temperaturasini kiriting12
21
2
3
4
34
23
22
32
43
44
23
soat18da birinchi bor manfiy temperatura bo'ldiyana ishlaysizmi ?1.ha:2.yo'q1
Dasturlar\азик\2.exe
1
```

227. 15 ta butun sonlardan iborat massiv tuzing va ulardan eng kichigini toping?

```
#include <iostream>
using namespace std;
```

```

int main(int argc, char *argv[]){
int a[15];
int i, mina, b;
cout<<"15 ta butun sondan iborat massiv kiriting";
for(i=1;i<15;i++)
cin>>a[i];
mina=a[1];
for(i=2;i<15;i++)
if(a[i]<mina)
mina=a[i];
cout<<"Massivning eng kichik elementi"<<mina;
return 0; }

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
15 ta butun sondan iborat massiv kiriting2
123
33
43
45
45
6
45
65
56
55
45
34
7
Massivning eng kichik elementi2Для продолжения нажмите любую клавишу . . . _

```

228. Elementlari 1, 2,..., 10 s da jismning yerga erkin tushish jarayonida bosib o'tilgan masofalardan iborat massivni shakllantiring.

```

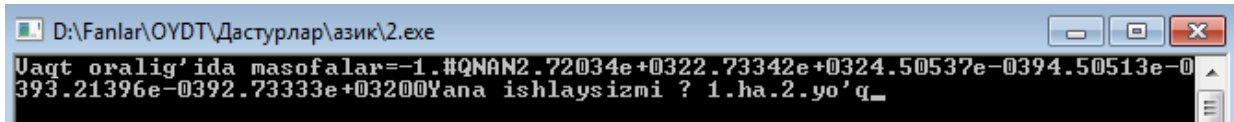
#include <iostream>
using namespace std;
int main(){
float const g=9.81;
float s[10];
float t[10];
int i,b;
L:
for(i=1;i<10;i++)
t[i]=i;
s[i]=g*t[i]*t[i]/2;
cout<<"Vaqt oralig'ida masofalar=";
for(i=1;i<10;i++)
cout<<s[i];

```

```

cout<<"Yana ishlaysizmi ? 1.ha.2.yo'q";
cin>>b;
if(b=1)
goto L;
return 0; }

```

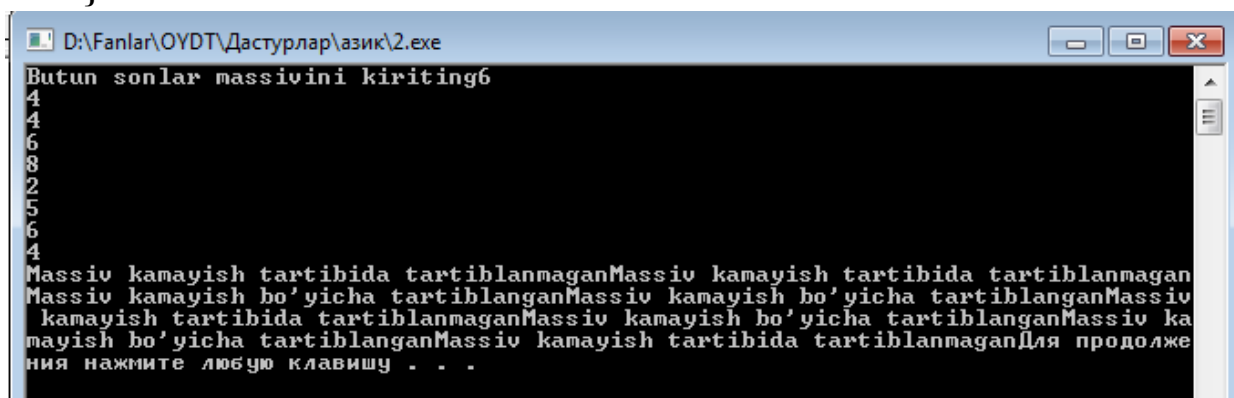


229. Butun sonlarning chiziqli massivi berilgan.Ularni kamayib borish tartibida joylashganligini tekshiring.

```

#include <iostream>
using namespace std;
int main(){
int a[10];
int i,b;
cout<<"Butun sonlar massivini kiriting";
for (i=1;i<10;i++)
cin>>a[i];
for(i=2;i<10;i++)
if(a[i-1]<a[i])
cout<<"Massiv kamayish bo'yicha tartiblangan";
else
cout<<"Massiv kamayish tartibida tartiblanmagan";
return 0;
}

```



230. Butun sonlarning chiziqli massividagi musbat elementlari yig'indisini toping. Massiv hajmi=10. Massivni klavitura oqali to'ldiring.

```

#include <iostream>
using namespace std;

```

```

int main(int argc, char *argv[]){
int a[10];
int s,i,b;
L:
cout<<"butun sonlar massivini tuzing";
for(i=1;i<10;i++)
cin>>a[i];
s=0;
for(i=1;i<10;i++)
if(a[i]>0)
s=s+a[i];
cout<<"musbat sonlar yig'indisi="<<s;
cout<<"yana ishlaysizmi?1.ha.2.yo'q";
cin>>b;
if(b=1)
goto L;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
butun sonlar massivini tuzing12
2
32
43
54
66
76
77
56
musbat sonlar yig'indisi=418yana ishlaysizmi?1.ha.2.yo'q_

```

231. Berilgan chiziqli massivdagi elementlarni o'sib borish tartibida joylashtiring.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int a[10];
int i,x,y,z,n,b;
L:
cout<<"Massiv o'lchovini kiriting";
cin>>n;
cout<<"Massiv elementlarini kiriting";
for(i=1;i<=n;i++)

```

```

cin>>a[i];
for(x=1;x<=n;x++)
for(y=1;y<=n;y++)
if(a[x]>a[y]){
z=a[x];
a[x]=a[y];
a[y]=z;}
for(x=1;x<n;x++)
cout<<a[x];
cout<<"yana ishlaysizmi ? 1.ha.2.yo'q";
cin>>b;
if(b==1)
goto L;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Massiv o'lchovini kiriting4
Massiv elementlarini kiriting3
2
5
6
653yana ishlaysizmi ? 1.ha.2.yo'q1
Massiv o'lchovini kiriting

```

232. Bemorning 1 sutka ichidagi temperatura ko'rsatgichlarini analiz qiluvchi dasturni tuzing: eng kichik va eng katta ko'rsatgichni, o'rta arifmetigini aniqlang. Temperatura sutkaga 6 marta o'lchanadi va natijalar klaviatura orqali T massivga kiritiladi.

```

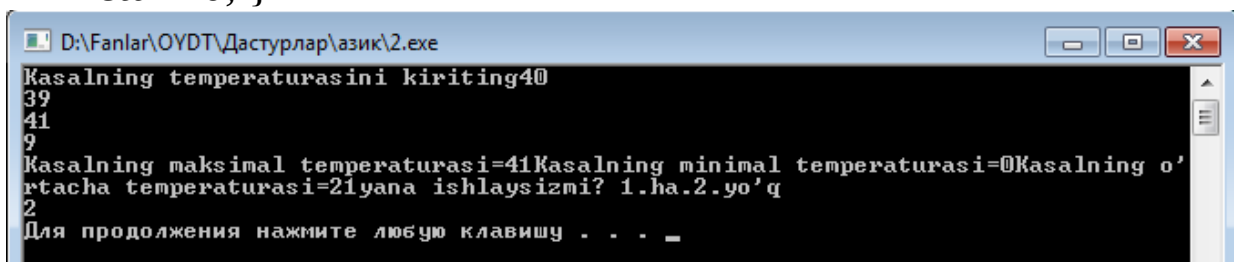
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int t[6];
int i,maxt,mint,st,b;
L:
cout<<"Kasalning temperaturasini kiriting";
for(i=2;i<6;i++)
cin>>t[i];
maxt=t[i];
for(i=2;i<6;i++)
if(t[i]>maxt)

```

```

maxt=t[i];
cout<<"Kasalning maksimal temperaturasi="<<maxt;
mint=t[1];
for(i=2;i<6;i++)
if(t[i]<mint)
mint=t[i];
cout<<"Kasalning minimal temperaturasi="<<mint;
st=0;
for(i=1;i<6;i++)
st=st+t[i];
st=st/6;
cout<<"Kasalning o'rtacha temperaturasi="<<st;
cout<<"yana ishlaysizmi? 1.ha.2.yo'q";
cin>>b;
if(b==1)
goto L;
return 0; }

```



```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Kasalning temperaturasini kiriting40
39
41
9
Kasalning maksimal temperaturasi=41Kasalning minimal temperaturasi=0Kasalning o'
rtacha temperaturasi=21yana ishlaysizmi? 1.ha.2.yo'q
2
Для продолжения нажмите любую клавишу . . .

```

233. Agar bir o'lchovli massivda faqatgina bitta manfiy va juft son bo'lsa ham t o'zgaruvchiga rost qiymat beradigan dastur tuzing.

```

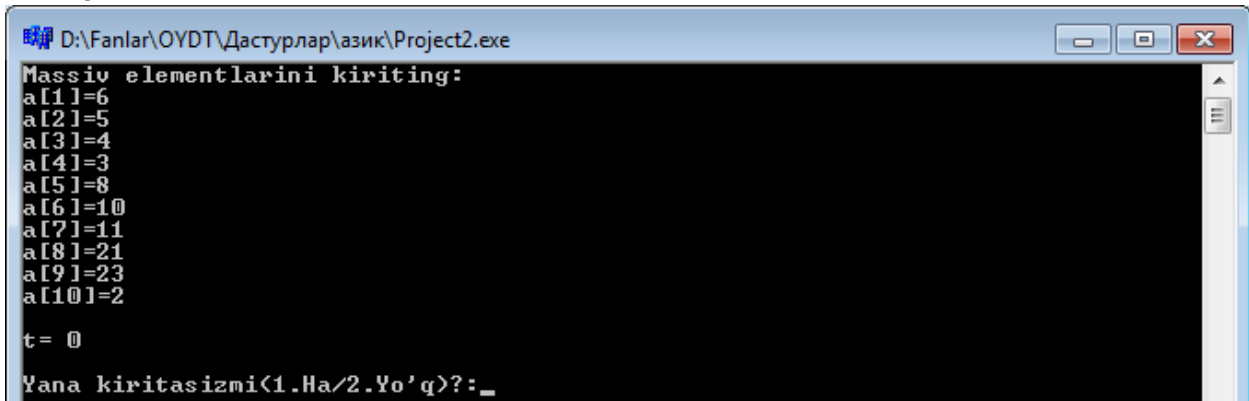
#include <iostream>
using namespace std;
int main(){
bool t=false;
int const n=10;
int k,a[n];
m2: cout<<"Massiv elementlarini kiriting:\n";
for (int i=1;i<=n;i++){
cout<<"a["<<i<<"]=";
cin>>a[i]; }
for (int i=1;i<=n;i++)
if (a[i]<0 && a[i]%2==0){
t=true;

```

```

break; }
cout<<"\nt= "<<t<<"\n";
cout<<"\nYana kiritasizmi(1.Ha/2.Yo'q)?:";
cin>>k;
if(k==1) goto m2;
return 0;
}

```



```

D:\Fanlar\OYDT\Дастурлар\азик\Project2.exe
Massiv elementlarini kiriting:
a[1]=6
a[2]=5
a[3]=4
a[4]=3
a[5]=8
a[6]=10
a[7]=11
a[8]=21
a[9]=23
a[10]=2

t= 0

Yana kiritasizmi(1.Ha/2.Yo'q)?:_

```

234. 30 ta xonadondan iborat uyda, yashovchilarni shunday ko'chiringki, birinchi xonadon egalari 30-xonadonga, 30-xonadondan esa 1-xonadonga, 2-dan 29-ga ko'chib o'tishsin va h.k, 5 ta kishidan ortiq odamlar turuvchi xonadonlar sonini aniqlang.

```

#include <iostream>
using namespace std;
int main(){
    const int n=31;
    int d,i,k,a[n-1];
    for(i=1;i<=n-1;i++){
        cout<<i<<"-xonadon kishilari soni:";
        cin>>a[i]; }
    d=(n-1)/2;
    for(i=1;i<=d;i++){
        k=a[i];
        a[i]=a[n-i];
        a[n-i]=k;}
    k=0;
    for (i=1;i<=n-1;i++)
        if(a[i]>5) k++;
    cout<<"Odamlari soni 5 tadan ko'p bo'lgan kvartiralar soni:
"<<k<<" ta\n"<<n-1<<" ta kvartiradagi odamlar soni:\n";

```

```

for (i=1;i<=n-1;i++)
cout<<i<<"-kvartirada: "<<a[i]<<"\n";
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
22-xonadon kishilari soni:5
23-xonadon kishilari soni:4
24-xonadon kishilari soni:3
25-xonadon kishilari soni:6
26-xonadon kishilari soni:7
27-xonadon kishilari soni:8
28-xonadon kishilari soni:9
29-xonadon kishilari soni:0
30-xonadon kishilari soni:8
Odamlari soni 5 tadan ko'p bo'lgan kvartiralar soni: 18 ta
30 ta kvartiradagi odamlar soni:
1-kvartirada: 8
2-kvartirada: 0
3-kvartirada: 9
4-kvartirada: 8
5-kvartirada: 7
6-kvartirada: 6
7-kvartirada: 3
8-kvartirada: 4
9-kvartirada: 5
10-kvartirada: 6
11-kvartirada: 7
12-kvartirada: 9

```

235. Ertak davlatda gaz, neft, olovni tejash maqsadida barcha yuklarni o'zining 1 dan 100 gacha bo'lgan bort nomeriga ega Zmey Gorinochlarda o'tkazishadi. Zmeyning boshlar soni uning yukni qanchalik darajada ko'tara olishini aniqlaydi. O'lmas kashshey arxivida har bir Zmey Gorinoch uchun shaxsiy ish ochilgan u yerda boshlar soni v abort nomeri haqidagi ma'lumotlar bor. Bir kuni Kalin ko'prigida Zmey Gorinoch tomonidan alvasti kampir urib yuborilibdi. Javobgar qochib qolibdi, lekin guvohlar shuni aytib o'tishibdi-ki, Zmeyning boshlar soni uning bort nomeriga karrali. Janob Nikitchinga yo'l harakatini buzgan Zmeyni aniqlashga yordam bering. Masalaning dasturi quyida keltirilgan.

```

#include <iostream>
using namespace std;
int main(){
const int n=100;
int i,a[n];
cout<<"Zmey Gorinochni boshlari sonini kiriting: ";
for(i=1;i<=n;i++)
cin>>a[i];
for(i=1;i<=n;i++)
if (i % a[i]==0)

```

```

    cout<<"\n"<<i<<" bortovoy nomerli Zmey Gorinich Yalmog'iz
kampirni urib ketgan bo'lishi mumkin.\n";
    return 0;
}

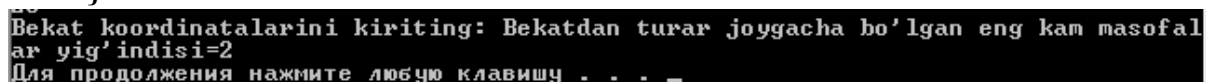
```

236. Bitta tumanda 30 ta mahalliy punktlar joylashgan. Tuman orqali temir yo'l o'tadi. Tuman aholisi iltimosiga binoan temiryo'l bekati qurilishi kerak. Ular yo'llarni shunday joylashtirishlari kerakki, agar tuman teritoriyasidan o'tuvchi yo'lning bir qismi to'g'ri chiziqli bo'lsa barcha mahalliy punktlardan bekatgacha bo'lgan masofa yig'indisini minimal bo'lishi kerak.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(){
    const int n=30;
    int i;
    double s,a[n],x[n],y[n];
    cout<<"Turar joyning koordinatalarini kiriting: ";
    for (i=1;i<=n;i++)
        cin>>x[i]>>y[i];
    cout<<"Bekat koordinatalarini kiriting: ";
    cin>>x[0]>>y[0];
    s=0;
    for (i=1;i<=n;i++){
        a[i]=pow(pow(x[i]-x[0],2)+pow(y[i]-y[0],2),1/2);
        s+=a[i];}
    cout<<"Bekatdan turar joygacha bo'lgan eng kam masofalar
yig'indisi="<<s<<endl;
    return 0;
}

```



237. Agar bir o'lchovli massivda 3 ta bir xil element ketma-ket joylashsa, r o'zgaruvchiga rost qiymati berilsin.

```

#include <iostream>
using namespace std;
int main(){

```

```

const int n=10;
int i,a[n+2];
bool r=false;
cout<<"Massiv elementlarini kiriting:\n";
for(i=1;i<=n;i++){
    cout<<"a["<<i<<"]: ";
    cin>>a[i];}
for(i=1;i<=n-2;i++)
    if(a[i]==a[i+1] && a[i]==a[i+2]){
        r=true;
        break;}
cout<<"r="<<r<<"\n";
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
Massiv elementlarini kiriting:
a[1]: 2
a[2]: 3
a[3]: 4
a[4]: 5
a[5]: 6
a[6]: 4
a[7]: 3
a[8]: 6
a[9]: 7
a[10]: 8
r=0
Для продолжения нажмите любую клавишу . . .

```

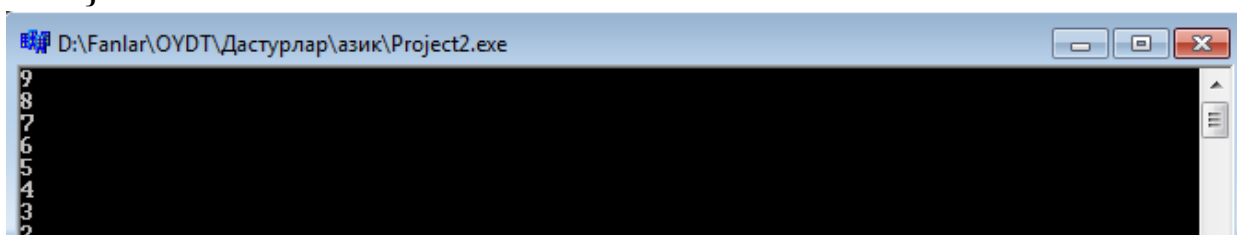
238. 20 ta elementdan iborat massivni kiriting va ulardan bir xil qiymatlar bor elementlar mavjudligini aniqlang.

```

#include <iostream>
using namespace std;
int main(){
    int a[19];
    int i,j,k;
    for(i=0;i<=19;i++)
        cin>>a[i];
    k=0;
    for(i=0;i<=19;i++)
        for(j=0;j<=19;j++)
            if (a[i]==a[j]) k++;
    cout<<"k="<<k<<endl;
    return 0;
}

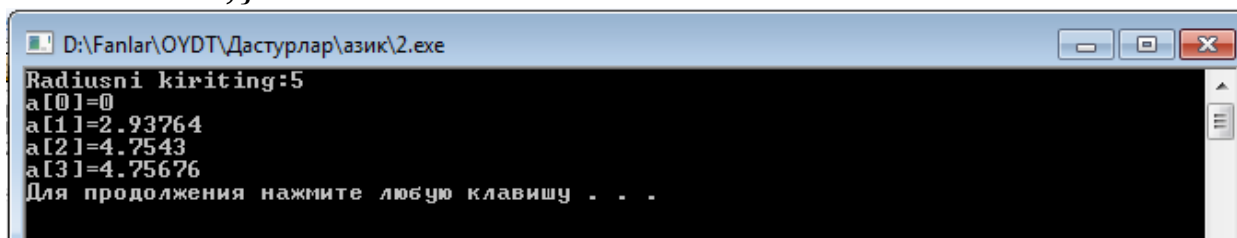
```

}



239. Tom qirqimi R m radiusli yarim aylana shakliga ega. Har $R/5$ m dan keyin oʻrnatiladigan tayanch ustunlaridan iborat jadval hosil qiling.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(){
    const double pi=3.14;
    double a[19];
    int r,i;
    cout<<"Radiusni kiriting:";
    cin>>r;
    for (i=0;i<=3;i++){
        a[i]=r*sin(i*pi/5);
        cout<<"a["<<i<<"]="<<a[i]<<endl;}
    return 0;}
```



240. Zavod ishlab chiqadigan mahsulot nomlarining jadvali berilgan. Bu jadvalda birinchi mahsulot nomi takrorlanayotganligini aniqlang, va agar takrorlansa birinchi mahsulot nomini jadvaldan oʻchirib tashlang.

```
#include <iostream>
#include <string.h>
using namespace std;
int main(){
    string t[19];
    int n,i;
    cout<<"Massiv oʻlchamini kiriting:";
```

```

cin>>n;
cout<<"Massiv elementlarini kiriting:\n";
for(i=0;i<19;i++)
cin>>t[i];
for(i=1;i<=19;i++)
if(t[0]==t[i]){
for(i=0;i<=18;i++){
t[i]=t[i+1];
cout<<"t["<<i<<"]="<<t[i]<<endl;}
break;}
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
2
3
4
4
5
4
4
3
4
4
1
0
1
2
1
1
3
2
1
3
4
5
5
6
7
2
6
5
4
3
3
3
4
5
4
3
4

```

241. N ta odamdan iborat mahsulot birjasining brokerlari familiyalari ro'yxati berilgan. Brokerlar familiyalarining o'rinlarini almashtiring: birinchisi va oxirgisini, ikkinchisi va oxirgidan bitta oldingisini, boshidan uchinchi va oxiridan uchinchisini va h.k.

```

#include <iostream>
#include <string.h>
using namespace std;
int main(){
const int n=31;
int d,i;
string k,a[n-1];
for(i=1;i<=n-1;i++){
cout<<i<<"-broker familiyasi:";
cin>>a[i];}
d=(n-1)/2;
for(i=1;i<=d;i++) {

```

```

k=a[i];
a[i]=a[n-i];
a[n-i]=k;}
for(i=1;i<=n-1;i++)
cout<<a[i];
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\2.exe
7-broker фамилия:ф
8-broker фамилия:в
9-broker фамилия:а
10-broker фамилия:п
11-broker фамилия:р
12-broker фамилия:о
13-broker фамилия:л
14-broker фамилия:д
15-broker фамилия:о
16-broker фамилия:рп
17-broker фамилия:пп
18-broker фамилия:аа
19-broker фамилия:вы
20-broker фамилия:чс
21-broker фамилия:см
22-broker фамилия:вы
23-broker фамилия:ув
24-broker фамилия:ув
25-broker фамилия:увв
26-broker фамилия:уу
27-broker фамилия:вв
28-broker фамилия:ы
29-broker фамилия:ввы
30-broker фамилия:вы

```

242. Sizning fakultetingiz kotibi ishini engillashtiruvchi dasturni tuzing. Siz kiritgan familiya orqali telefonni qidiruvchi dastur tuzing.

```

#include <iostream>
#include <string>
using namespace std;
int main(){
string a[19],b[19],c;
int n,i;
cout<<"Massiv o'lchamini kiriting:";
cin>>n;
cout<<"Familiya va telefon kiriting:\n";
for(i=0;i<n;i++)
cin>>a[i]>>b[i];
cout<<"Telefon raqami kerak bo'lgan familiyani kiriting:\n";
cin>>c;
for(i=0;i<n;i++)
if(strcmp(a[i],c)==0){
cout<<"Siz qidirgan telefon raqam: " << b[i]

```

```

<<endl;
break;}
return 0;
}

```

243. Ikkita jadval berilgan. Bittasida xizmat nomlari, ikkinchisida bu xizmat uchun to‘lash kerak bo‘lgan pul miqdori yozilgan. 2 ta jadvalda ham P narxga teng bo‘lgan xizmatlarni o‘chirib tashlang.

```

#include <iostream>
#include <string.h>
using namespace std;
int main(){
string a[19];
int b[19];
int i,p,n,k;
cout<<"Massiv o‘lchovini kiriting:"<<endl;
cin>>n;
for(i=0;i<n;i++) {
cout<<"a[ "<<i+1<<" ]="; cin>>a[i]; }
for(i=0;i<n;i++) {
cout<<"b[ "<<i+1<<" ]="; cin>>b[i]; }
cout<<"Xizmat pulini kiriting:"<<endl;
cin>>p;
for(i=0;i<n;i++)
if(b[i]==p){
k=i;
for(i=k;i<n-1;i++){
a[i]=a[i+1];
b[i]=b[i+1];}
for(i=0;i<n-1;i++)
cout<<"b[i]="<<b[i]<<" a[i]="<<a[i]<<endl;
}
return 0; }

```

```
C:\Program Files\Borland\CBUILDER6\Projects\Project1.exe
Massiv o'lchovini kiriting:
5
a[ 1 ]=6
a[ 2 ]=7
a[ 3 ]=8
a[ 4 ]=9
a[ 5 ]=3
b[ 1 ]=4
b[ 2 ]=5
b[ 3 ]=4
b[ 4 ]=3
b[ 5 ]=4
Xizmat pulini kiriting:
3
b[i]=4 a[i]=6
b[i]=5 a[i]=7
b[i]=4 a[i]=8
b[i]=4 a[i]=3
```

Mustaqil ishlash uchun vazifa

244. A[1..8] massivni elementlari -10 dan to 10 gacha bo'lgan tasodifiy sonlar generatori yordamida tuzing va uni ekranga chiqaring. Massivdagi manfiy sonlar miqdorini hisoblang.

245. A[1...12] massivni -20 dan to 10 gacha bo'lgan elementlar ishtirok etgan tasodifiy sonlar generatori yordamida tuzing va uni ekranga chiqaring. Massivdagi manfiy sonlarni 0 soni bilan almashtiring.

246. A[1...15] massivni -15 dan to 30 gacha bo'lgan elementlar ishtirok etgan tasodifiy sonlar generatori yordamida tuzing va uni ekranga chiqaring. Massivdagi eng katta elementni va uning indeksini aniqlang.

247. Noyabr dekadasining havo temperaturasi haqidagi ma'lumotlar massivda saqlanadi. Temperatura necha marta -10 gradusdan past bo'lganini aniqlang.

248. Qora dengiz sohillaridagi havo temperaturasi haqidagi sentabr dekadasining ma'lumotlari massivda saqlanadi. Bu vaqt ichida necha kun cho'milishga yaroqli ekanligini aniqlang.

249. Aprel dekadasining havo temperaturasi va yog'inlar miqdori haqidagi ma'lumotlar massivda saqlanadi. Yog'inlar necha marotaba yomg'ir va necha marotaba qor ko'rinishda yog'anligi aniqlang.

250. Dekabr dekadasini uchun havo temperaturasi haqidagi ma'lumotlar massivda saqlanadi. Temperatura hecha marotaba bu dekadadagi o'rtacha temperaturadan katta bo'lganligi aniqlang.

251. Noyabr dekadasidagi shamol yo'nalishi (shimoliy, janubiy, g'arbiy, sharqiy) va shamol kuchi haqidagi ma'lumotlar massivda

saqlanadi. Janubiy shamol necha kun 8 m/s dan oshuvchi kuch bilan esdi?

252. Butun sonlar massivining juft elementlar aniqlang. Massivi hajmi 20. Massivi 100 dan to 200 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

253. Butun sonlar massivida 7 ga karrali sonlarning ko'paymasini toping. Massiv kattaligi -15. Massivni 10 dan to 50 gacha bo'lgan tasodiy sonlar bilan to'ldiring.

254. Butun sonlar massividagi toq nomerlarga ega bo'lgan elementlar yig'indisini toping. Massiv kattaligi-20. Massivni 100dan to 200 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

255. Butun sonli massivdagi 0 dan kichik barcha elementlar ko'paytmasi topilsin. Massiv kattaligi-10. Massivni klaviatura orqali to'ldiring.

256. Butun sonlardan iborat massivning quydagi shartga javob beruvchi barcha elementlar yig'indisini aniqlang: 4ga bo'lganda qoldiq 3 ga teng. Massiv kattaligi -20. Massivni 200 dan to 300 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

257. Butun sonli massivdagi kiritilgan sonlardan katta bo'lgan barcha elimentlar yig'indisini toping. Massiv kattaligi -20. Massivni 50 dan to 100 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

258. Butun sonli massivdagi kiritilgan sonlardan kichik bo'lgan barcha elimentlar ayirmasini toping. Massiv kattaligi -10. Massivni 50 dan to 100 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

259. 3 ga 9 ga karrali massiv elimentlarining ko'paytmasini toping. Massiv kattaligi -10. Massivdagi 5 dan to 500 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

260. Butun sonlardan iborat massivdagi massiv elimentlarining o'rta arifmetigidan kichik bo'lgan barcha elimentlar yig'indisi hisoblansin.

261. Butun sonlar massividagi ham 5 ga ham 8 ga bo'linadigan elimentlar yig'indisini toping. Massiv kattaligi -30. Massivni 500 dan to 1000 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

262. Butun sonlarning chiziqli massivida 5 ga karrali elimentlarning ko'paytmasini toping. Massiv kattaligi -10. Massivni 10 dan 100 gacha bo'lgan tasodifiy sonlar bilan to'ldiring.

263. Chiziqli massiv bir yilda 12 oydagi yog'inlar miqdori haqidagi ma'lumotni o'z ichida saqlaydi. Bu yildagi umumiy yog'inlar miqdorini, oydagi o'rtacha miqdorini, qurg'ochilikka boy oylarni (yog'inlar miqdori 30 mm dan kam bo'lganda), yilning eng qurg'oqchil oyini aniqlovchi dasturni tuzing.

264. Bir o'lchovli massivda juft elementlar sonini aniqlang.

265. Bir o'lchovli massivda oldindan a songa teng bo'lgan birinchi son uchrashguncha juft sonlar sonini aniqlang.

266. Bir o'lchovli massivda tartib bo'yicha eng kichik elementdan oldin joylashgan elementlarning o'rta arifmetigi hisoblansin.

267. Bir o'lchovli massivda -2 dan 10 gacha bo'lgan oraliqqa kiruvchi elementlarning o'rta arifmetigi hisoblansin.

268. Bir o'lchovli massivdagi birinchi uchragan 0 lik elementgacha barcha juft elementlar yig'indisi hisoblansin.

269. Agar bir o'lchovli massivda eng katta element yagona bo'lsa va oldindan berilgan a sondan oshmasa t o'zgaruvchiga rost qiymati berilsin.

270. Bir o'lchovli massivda $j * j < a_j < i!$ tengsizlik bajariladigan elementlar sonini aniqlang.

271. 2 ta a va b natural sonlari berilgan. Agar massivda faqatgina 1 ta a ga karrali bo'lib turib b ga karrali bo'lmagan son mavjud bo'lsa w o'zgaruvchisiga rost qiymati berilsin.

272. Bir o'lchovli massivda o'zining nomeriga teng, shu jumladan 3 ga karrali bo'lgan elementlar miqdorini aniqlang.

273. a bir o'lchovli massiv berilgan. A massivdagi faqatgina o'zining nomeridan 10 taga katta bo'lgan elementlardan iborat yangi massiv tuzing. Agar bunday elementlar bo'lmasa, shu haqda ma'lumot chiqarilsin.

274. Bir o'lchovli massivda juft nomerga ega bo'lgan elementlardan eng katta elementini toping. Uning yagona ekanligini aniqlang.

275. Chiziqli massivni minimal elementga yaqin bo'lgan elementlarni olib tashlash yo'li bilan qisqartiring.

276. A(1;20) massiv berilgan. Uning barcha no'llik elementlar soni topilsin.

277. $X(1;n)$ massivida har bir element 0, 1 yoki 5 ga teng. Massivni shunday joylashtiring-ki, birinchi barcha no'llar, so'ngra esa 5 lar joylashsin. Qo'shimcha massiv tuzilmasin.

278. Chiziqli massivda shunday ikkita element toping-ki, ularning yig'indisi maksimal bo'lsin.

279. O'zbekiston futbol jamoalarinig ro'yhati berilgan va O'zbekiston chempionatida har bir jamoa to'plagan ballari berilgan. Ma'lumki, bir xil ball to'plagan jamoalar yo'q, 2 ta eng kam ball to'plagan jamoalar esa oliy ligani tark etishadi. Bu qaysi jamoalar?

280. Ikkita chiziqli A va B jadvallarni yangi C jadvalga joylashtiring, bunda A jadval elementlarini toq o'rinlarga, B jadval elementlarini esa juft o'rinlarga o'rnatish.

281. Lotereyada 100 ta chipta o'ynaldi. Jadval 10 ta yutuqli chipta nomerlariga ega. N nomerli chipta yutuqli bo'l'shini tekshiring.

282. Chiziqli massivda eng katta element topilsin. Elementning tartib raqamini yoniga qo'ying, qolgan barcha elementlarni bitta pozitsiya o'ngga suring.

283. Oylikka berilgan ma'lumotnoma 2 ta jadval ko'rinishida tasvirlangan. Bittasida tsex ishchilarining familiyalari, 2-sida esa ularning ushbu oydagi oylik maoshi ko'rsatilgan. Oyligi ushbu oydagi barcha ishchilarning o'rtacha oyligidan eng kam farq qiluvchi ishchining familiyasini toping.

284. 2 ta bir xil kattalikka ega bo'lgan chiziqli massiv berilgan. Bir xil indeksli joylarda turuvchi 2 ta massiv elementlarining ayirmasidan uchinchi massivni hosil qiling.

285. 10 ta elementdan iborat sonlardan tuzilgan chiziqli massiv berilgan. O'sib boruvchi ketma-ketlik qolishi uchun $A[1]$, $A[2]$, ..., $A[10]$ ketma-ketlikdan olib tashlash kerak bo'lgan eng kam K elementlar sonini toping.

286. Butun sonlarning bir o'lchovli massivi berilgan. Bu massivda, eng faol son necha marta takrorlanishini aniqlang.

287. 10 ta butun sondan iborat bir o'lchovli massiv berilgan. Undagi turli xil sonlarning miqdorini aniqlang.

288. 10 ta butun sondan iborat bir o'lchovli massiv berilgan. Undagi sonlarni ketma-ket joylashgan bir xillarining miqdorini aniqlang.

ANIQ INTEGRALLARNI HISOBLASH

Qisqa nazariy kirish. Ko‘p hollarda ilmiy-texnikaviy masalalarni yechishda aniq integralni yoki boshlang‘ich funksiya qiymatini hisoblashga to‘g‘ri keladi. Ayrim ta’riflarni eslatib o‘tamiz.

1. Agar oraliqning barcha nuqtalarida $F(x)$ funksiyaning hosilasi $f(x)$ bo‘lsa, $F(x)$ funksiya berilgan oraliqda $f(x)$ funksiya uchun boshlang‘ich deb ataladi, ya’ni

$$F'(x) = f(x)$$

2. $f(x)$ funksiya ma’lum bir $[a, b]$ kesmada berilgan bo‘lsin. Bu kesmani ixtiyoriy ravishda $\Delta x_i = x_{i+1} - x_i$ ($i = 0, 1, 2, \dots, n-1$) qismlarga bo‘lib tashlaylik. Har bir qismdan ixtiyoriy ξ_i ($i = 0, 1, 2, 3, \dots, n-1$) nuqtani olamiz. Bunda a dan b gacha

bo‘lgan kesmada joylashgan $f(x)$ funksiyaning $\int_a^b f(x)dx$ aniq integrali deb quyidagiga aytiladi

$$\lim_{\max \Delta x_i \rightarrow 0} \sum_{i=0}^{n-1} f(\xi_i) \Delta x_i$$

3. Agar $[a, b]$ ni o‘z ichiga oluvchi oraliqda $F(x)$ $f(x)$ uchun boshlang‘ich funksiya bo‘lsa, unda

$$\int_a^b f(x)dx = F(b) - F(a).$$

4. Agar $\int_a^b f(x)dx$ mavjud bo‘lsa, unda $[a, b]$ da $f(x)$ uchun

boshlang‘ich funksiylardan biri bo‘lib $\int_a^x f(t)dt$ olinadi.

Shunday qilib, boshlang‘ich funksiyaning hisoblashni bilgan holda, biz aniq integralni hisoblab topishimiz va aksincha qilishimiz mumkin. Lekin odatda boshlang‘ich funksiyaning elementar funksiya orqali ifodalash har doim mumkin emas. Shuning uchun taqribiy integratsiyalashdan foydalanishga to‘g‘ri keladi. Masalani hal etish

uchun ko'plab sonli usullar mavjud, biz ulardan ikkitasini ko'rib chiqamiz, bular: trapetsiya usuli va Simpson usuli.

Trapetsiya usuli. Ma'lumki $\int_a^b f(x)dx$ aniq integral kattaligi

o'zidan $f(x)$ funksiya grafigi, absissalar o'qi va 2 ta $x=a$ va $x=b$ to'g'ri chiziqlar bilan chegaralangan egri chiziqli trapetsiya yuzini namoyish qiladi.

$[a, b]$ integrallashtirish kesmasini $h=(b-a)/n$ uzunlikdagi n ta bir xil qismga bo'laylik. $x_0=a, x_1=a+h, \dots, x_n=b$ bo'linish nuqtalarida $y=f(x)$ egri chiziq, ya'ni $y_i=f(x_i), x_i=a+ih, i=0,1,\dots,n$ bilan kesishgunga qadar $y_0, y_1, \dots, y_n$ ordinatalarini o'tkazamiz. Shunda bu figuraning yuzi (uni biz S orqali ifodalaymiz) trapetsiyalar yuzlarining yig'indisiga teng.

$$S = h \left(\frac{y_0 + y_1}{2} + \frac{y_1 + y_2}{2} + \dots + \frac{y_{n-1} + y_n}{2} \right) = \\ = \frac{b-a}{2n} (y_0 + 2y_1 + \dots + 2y_{n-1} + y_n).$$

Shunday qilib, trapetsiya formulasi orqali integralning yaqinlashtirilgan qiymati quyidagi ko'rinishda yoziladi.

$$\int_a^b f(x)dx \approx \frac{b-a}{2n} (y_0 + 2y_1 + \dots + 2y_{n-1} + y_n).$$

Simpson usuli. $[a, b]$ integrallashtirish kesmasini $h=(b-a)/2n$ uzunlikdagi $2n$ ta teng qismlarga bo'lamiz. $x_0=a, x_1, x_2, \dots, x_{2n}=b$ bo'linish nuqtalari $y_0, y_1, y_2, \dots, y_{2n}$ integral osti qiymati, ya'ni $y_i=f(x_i), x_i=a+ih, i=0,1,\dots,2n$ ga mos kelsin. $[x_0, x_2]$ kesmada $f(x)$ integral osti funksiyasi $(x_0, y_0), (x_1, y_1), (x_2, y_2)$ nuqtalardan o'tuvchi parabola bilan almashtiramiz. Parabola tenglamasini Nuyutonning interpolatsiyon formulasi ko'rinishida yozib olamiz:

$$f(x) = f(x_0) + \tilde{x} \Delta f(x_0) + \frac{\tilde{x}(\tilde{x}-1)}{2} \Delta^2 f(x_0) = \\ = y_0 + \tilde{x} \Delta y_0 + \frac{\tilde{x}(\tilde{x}-1)}{2} \Delta^2 y_0$$

Bu yerda

$$\tilde{x} = \frac{x - x_0}{h}; \quad \Delta y_0 = y_1 - y_0;$$

$$\Delta^2 y_0 = \Delta y_1 - \Delta y_0 = y_2 - 2y_1 + y_0$$

Hosil qilingan ifodadan integrallashtirib quyidagiga ega bo‘lamiz

$$\int_{x_0}^{x_2} f(x)dx = \int_{x_0}^{x_2} \left\{ y_0 + (y_1 - y_0)\tilde{x} + (y_2 - 2y_1 + y_0)\frac{\tilde{x}(\tilde{x}-1)}{2} \right\} = \frac{h}{3}(y_0 + 4y_1 + y_2)$$

Huddi shu kabi,

$$\int_{x_2}^{x_4} f(x)dx = \frac{h}{3}(y_2 + 4y_3 + y_4),$$

.....

$$\int_{x_{2n-2}}^{x_{2n}} f(x)dx = \frac{h}{3}(y_{2n-2} + 4y_{2n-1} + y_{2n}).$$

Ushbu tenglamalarni qo‘shib, Simpson formulasi nomi bilan mashxur formulaga ega bo‘lamiz:

$$\int_a^b f(x)dx \approx \frac{b-a}{6n}(y_0 + 4y_1 + 2y_2 + \dots + 4y_{2n-1} + y_{2n}).$$

Misollar

289. Integrallashtirish orqali $n=30$ qismga bo‘lib, trapetsiya

usuli bilan $\int_0^{\pi/2} \cos x dx$ ni toping. $\pi/10$ qadam bilan $[0, \pi/2]$ kesmada

boshlang‘ich funksiya qiymatini har bir kesmani yana 10 ta qismga bo‘lish yo‘li orqali Simpson usuli bilan hisoblab toping. Boshlang‘ich qiymatlarini massiv ko‘rinishida ifodalang. Bir xil nuqtalarda hisoblashni nazorat qilish uchun ($y=\sin(x)$) boshlang‘ichning aniq qiymatlarini hisoblab toping va massiv ko‘rinishida ifodalang.

Berilgan masalada quyidagi qiymatlar chop etilishi kerak:

$$\int_0^{\pi/2} \cos x dx$$

ga teng son (dasturda S bilan belgilangan) hamda 3 ta

massiv. Birinchi massiv- x ning $\pi/10$ qadam bilan 0 dan to $\pi/2$ gacha bo'lgan qiymati (dasturda bu massiv X orqali ifodalangan), ikkinchi massiv-Simpson usuli orqali olingan boshlang'ich qiymati (dasturda Z orqali ifodalangan), uchinchi massiv-boshlang'ichning aniq qiymatlari (dasturda Y orqali ifodalangan). $\pi/10$ qadam bilan 0 dan $\pi/2$ gacha o'zgaruvchi x da biz uning 6 ta qiymatini olamiz. X ning har bir qiymatida z va y larning qiymati mos keladi, shuning uchun 3 ta massiv 6 tadan komponentlarga ega bo'ladi.

Aniq integralni hisoblash uchun quyidagi kattalik ishlatiladi:

$$h_1 = \frac{\pi/2 - 0}{30} = \frac{\pi}{60},$$

bu yerda h_1 – integrallashtirish qadami;

$$x_i = ih_1,$$

$$y_i = \cos x_i, i = 0, 1, \dots, 30.$$

Dasturni tuzish paytida
$$\int_a^b f(x) dx \approx \frac{b-a}{2n} (y_0 + 2y_1 + \dots + 2y_{n-1} + y_n).$$

trapetsiya formulasini quyidagi ko'rinishda ifodalash maqsadga muvofiq bo'ladi

$$\int_0^{\pi/2} \cos x dx \approx \frac{h_1}{2} \left(y_0 + y_n + \sum_{i=1}^{n-1} 2y_i \right)$$

Yoki, $h_1 = \frac{\pi}{60}$ va $n=30$ ekanligini hisobga olgan holda quyidagiga ega bo'lamiz:

$$\int_0^{\pi/2} \cos x dx \approx S = \frac{\pi}{120} \left(y_0 + y_{30} + \sum_{i=1}^{29} 2y_i \right)$$

bu yerda

$$y_0 = \cos 0 = 1, \quad y_{30} = \cos \frac{\pi}{2} = 0, \quad y_i = \cos x_i,$$

yoki

$$S = \frac{\pi}{120} \left(1 + \sum_{i=1}^{29} 2 \cos x_i \right),$$

bu yerda

$$x_i = x_{i-1} + \pi / 60, \quad x_0 = 0, \quad i = 1, 2, \dots, 29.$$

Quyida keltirilgan algoritm asosida tuzilgan dastur keltirilgan.

```
#include <cstdlib>
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]) {
    const float pi=3.14;
    float x[30];
    x[0]=0;
    int i;
    float s=0;
    for(i=1;i<=29;i++) {
        x[i]=x[i-1]+pi/60;
        s+=cos(x[i]);
    }
    s=pi*(1+2*s)/120;
    cout<<"s= "<<s<<endl;
    return 0;
}
```



290. $\int_0^{\pi/2} \sqrt{1 - \frac{1}{4} \sin^2 x} dx$ ni integrallashtirish oralig'ini $2n=20$ ta

qismga bo'lish yo'li orqali Simpson usuli bilan toping.

Aniq integralni hisoblash uchun bizda quyidagilar mavjud.

$$h_1 = \frac{\pi/2 - 0}{2n} = \frac{\pi}{40},$$

$$x_i = ih_1,$$

$$y_i = \sqrt{1 - \frac{1}{4} \sin^2 x_i}, \quad i = 0, 1, 2, \dots, 20$$

Dasturni tuzish jarayonida Simpson formulasi

$$\int_a^b f(x) dx \approx \frac{b-a}{6n} (y_0 + 4y_1 + 2y_2 + \dots + 4y_{2n-1} + y_{2n}).$$

ni quyidagi ko'rinishda ifodalash maqsadga muvofiq bo'ladi.

$$\frac{h_1}{3} \left(y_0 + y_{20} + \sum_{i=1}^{2n-1} (3 + c_i) y_i \right),$$

bu yerda

$$c_i = \begin{cases} +1, & \text{agar } i \text{ toq bo'lsa,} \\ -1, & \text{agar } i \text{ juft bo'lsa,} \end{cases}$$

$$h_1 = \frac{\pi}{40}, \quad 2n=20 \text{ ni hisobga olgan holda biz quyidagiga ega}$$

bo'lamiz

$$\int_0^{\pi/2} \sqrt{1 - \frac{1}{4} \sin^2 x} dx \approx S = \frac{\pi}{120} \left(y_0 + y_{20} + \sum_{i=1}^{19} (3 + c_i) y_i \right),$$

bu yerda

$$y_0 = \sqrt{1 - \frac{1}{4} \sin^2 0} = 1, \quad y_{20} = \sqrt{1 - \frac{1}{4} \sin^2 \frac{\pi}{2}} = \frac{\sqrt{3}}{2}$$

yoki

$$S = \frac{\pi}{120} \left(y_0 + y_{20} + \sum_{i=1}^{19} (3 + c_i) \sqrt{1 - \frac{1}{4} \sin^2 x_i} \right),$$

bu yerda

$$c_i = -c_{i-1}, \quad c_1 = 1, \quad i = 2, 3, \dots, 19,$$

$$x_i = x_{i-1} + \pi/40, \quad x_0 = 0, \quad i = 1, 2, \dots, 19.$$

Quyida keltirilgan algoritm asosida dastur keltirilgan.

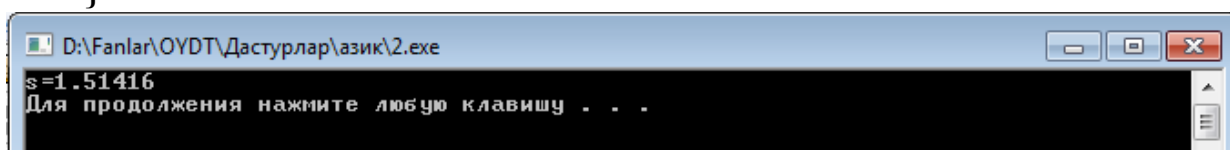
```
#include <cstdlib>
```

```
#include <iostream>
```

```

#include <math.h>
using namespace std;
#define pi 3.14
int main(int argc, char *argv[]) {
float c[20],x[20];
float s=0;
int i;
c[0]=1; x[0]=1;
for(i=1;i<=19;i++) {
c[i]=(-1)*c[i-1];
x[i]=x[i-1]+pi/40;
s+=(3+c[i])*sqrt(1-1/4*pow(sin(x[i]),2));
}
s=pi/120*(1+sqrt(3)/2+s);
cout<<"s="<<s<<endl;
return 0;
}

```



Mustaqil ishlash uchun vazifa

291. $[a, b]$ integrallashtirish oralig‘ini berilgan (n) sonli qismlarga bo‘lish yo‘li orqali, aniq integralni variantda ko‘rsatilgan usul (quyida keltirilgan jadvalga qarang) bilan hisoblash uchun algoritim va dasturni tuzing. Dasturda boshlang‘ichga nisbatan aniq integralning qiymatini hisoblashga alohida e’tibor bering.

	$F(x)$ Integral osti funk- siyasi	$[a,b]$ integra l oralig‘i	Integ- ralni aniqlash ning sonli usuli	Bo‘li- nish qismi soni	$\int_a^b f(x)dx$ boshlan- g‘ichning aniq qiymati
	$\frac{\ln x}{x\sqrt{1+\ln x}}$	$[1;3,5]$	Simpson	30	$\frac{2}{3}(\ln x + 1)^{\frac{3}{2}} - 2(\ln x + 1)^{\frac{1}{2}} + \frac{3}{4}$

	$\frac{1}{x \ln x}$	[2;3]	Simpson	36	$2,3026(\ln \ln x - \ln \ln 2)$
	$\frac{\ln^2 x}{x}$	[1;4]	Trapet-siya	52	$\frac{1}{3} \ln^3 x$
	$\sqrt{e^x - 1}$	[0;ln 2]	Simpson	104	$2\sqrt{e^x - 1} - 2 \arctg \sqrt{e^x - 1}$
	$x e^x \sin x$	[0;1]	Trapet-siya	48	$\frac{x e^x (\sin x - \cos x) + e^x \cos x - 1}{2}$
	$Xshx$	[0;2]	Simpson	48	$\frac{x(e^x + e^{-x})}{2} - \frac{e^x - e^{-x}}{2}$
	$\frac{1}{\sqrt{9 + x^2}}$	[0;2]	Trapet-siya	208	$\ln(x + \sqrt{x^2 + 9}) - \ln 3$
	$\frac{1}{\sqrt{x^2}} \sin \frac{1}{x}$	[1;2;5]	Simpson	44	$\cos \frac{1}{x} - \cos 1$
	$x \arctg x$	[0; $\sqrt{3}$]	Trapet-siya	48	$\frac{x^2}{2} \arctg x - \frac{x}{2} + \frac{1}{2} \arctg x$
	$\arcsin \sqrt{\frac{x}{1+x}}$	[0;3]	Simpson	36	$x \arcsin \sqrt{\frac{x}{1+x}} - \sqrt{x} + \arctg \sqrt{x}$
	$x^x (1 + \ln x)$	[1;3]	Trapet-siya	40	$x^x - 1$
	2^{3x}	[0;1]	Simpson	240	$\frac{1}{3 \ln 2} (2^{3x} - 1)$
	$\frac{x \arctg x}{\sqrt{1+x^2}}$	[0;1]	Trapet-siya	22	$\sqrt{1+x^2} \arctg x - (x + \sqrt{1+x^2})$
	$\frac{e^{3x} + 1}{e^x + 1}$	[0;2]	Simpson	48	$\frac{e^{2x}}{2} - e^x + x + 0,5$
	$\sin^2 x$	$[0; \frac{\pi}{2}]$	Trapet-siya	22	$\frac{x}{2} - \frac{1}{4} \sin 2x$
	$x^2 \sqrt{4 - x^2}$	[0; 1, 9..]	Simpson	96	$2 \arcsin \frac{x}{2} - \frac{1}{2} \sin(4 \arcsin \frac{x}{2})$

$e^x \cos^2 x$	$[0; \pi]$	Trapet– siya	60	$\frac{e^x}{2} \left(1 + \frac{2 \sin 2x + \cos 2x}{5} \right) - 0,6$
$(x \ln x)^2$	$[1; e]$	Simpson	52	$\frac{x^3}{27} (9 \ln^2 x - 6 \ln x + 2) - \frac{2}{27}$
$\arcsin \sqrt{\frac{x}{1+x}}$	$[0; 3]$	Trapet– siya	176	$x \arcsin \sqrt{\frac{x}{1+x}} - \sqrt{x + \arctg \sqrt{x}}$
$\frac{x^2 - 1}{(x^2 + 1) \sqrt{x^4 + 1}}$	$[0; 1]$	Simpson	36	$-\frac{\sqrt{2}}{2} \arcsin \left(\frac{\sin(2 \arctg x)}{\sqrt{2}} \right)$
$\frac{x^2 - 1}{(x^2 + 1) \sqrt{x^4 + 1}}$	$[0; 1]$	Trepet– siya	52	$\ln(\tg \frac{x}{2}) - (\cos x)(\ln(\tg x)) - \text{Intg} 0,5 + (\cos)$
$\frac{e^x (1 + \sin x)}{1 + \cos x}$	$[0; 1,5]$	Simpson	132	$\text{extg} \frac{x}{2}$
$\frac{1}{(3 \sin x + 2 \cos x)}$	$[0; 1]$	Simpson	78	$\frac{3}{26} - \frac{3 \cos x - 2 \sin x}{13(2 \cos x + 3 \sin x)}$
$\frac{x}{x^4 + 3x^2 + 2}$	$[1; 2]$	Trapet– siya	36	$\frac{1}{2} \ln \frac{x^2 + 1}{x^2 + 2} - \frac{1}{2} \ln \frac{2}{3}$

DIFFERENSIAL TENGLAMALARNI YECHISH

Qisqa nazariy kirish. Turli xil amaliy masalalarni yechishda ko‘p hollarda oddiy differensial tenglamalar uchrab turadi.

1. Oddiy differensial tenglama deb quyidagi ifodaga aytiladi:

$$F(x, y, y', y'', \dots, y^{(n-1)}, y^{(n)}) = 0,$$

bu yerda x – erkin o‘zgaruvchi, y – berilgan funksiya (x ga nisbatan), $y, y', y'', \dots, y^{(n-1)}, y^{(n)}$ – $1, 2, \dots, n-1, n$ tartibli hosila.

$F(x, y, y', y'', \dots, y^{(n-1)}, y^{(n)}) = 0$ tenglamaga kiruvchi hosilaning yuqori tartibi differensial tenglamaning tartibi deb ataladi.

2. Agar $y=f(x)$ ni $F(x, y, y', y'', \dots, y^{(n-1)}, y^{(n)}) = 0$ ga qo‘yganda oxirgi ayniyatga aylansa, unda $y=f(x)$ funksiya $F(x, y, y', y'', \dots, y^{(n-1)}, y^{(n)}) = 0$ tenglamaning yechimi bo‘ladi.

Har bir differensial tenglama umumiy qilib aytganda, cheksiz ko‘p yechimlarga ega. Shuning uchun hususiy yechimni toppish uchun boshlang‘ich shartni, aniqrog‘i $x = x_0$ bo‘lganda $y, y', y'', \dots, y^{(n-1)}$ qiymatni ko‘rsatishimiz kerak, ya’ni

$$y_0 = y(x_0), y'_0 = y'(x_0), \dots, y_0^{(n-1)} = y^{(n-1)}(x_0)$$

3. $F(x, y, y', y'', \dots, y^{(n-1)}, y^{(n)}) = 0$ tenglama yuqori hosilaga nisbatan yechilgan deyiladi, agar u quyidagi ko‘rinishga ega bo‘lsa

$$y^{(n)} = f(x, y', y'', \dots, y^{(n-1)})$$

$y_0 = y(x_0), y'_0 = y'(x_0), \dots, y_0^{(n-1)} = y^{(n-1)}(x_0)$ ko‘rinishdagi boshlang‘ich shartlarda $y^{(n)} = f(x, y', y'', \dots, y^{(n-1)})$ tenglamaning yechimlarini qidirish masalasi oddiy differensial tenglama uchun Koshi masalasi deb ataladi. $y^{(n)} = f(x, y', y'', \dots, y^{(n-1)})$ tenglamasi y' ni no‘malum $p_1(x)$ funksiya bilan, y'' ni $p_2(x), \dots, y^{(n-1)}$ ni $p_{n-1}(x)$ funksiya bilan almashtirish yo‘li orqali birinchi tartibli oddiy differensial tenglamalarning n sistemasiga keltiriladi. Shunday qilib quyidagiga ega bo‘lamiz.

$$y' = p_1,$$

$$p_1' = p_2,$$

$$p_2' = p_3,$$

.....

$$p_{n-1}' = f(x, y, p_1, p_2, \dots, p_{n-1})$$

jumladan

$$y(x_0) = y_0,$$

$$p_1(x_0) = y_0',$$

$$p_2(x_0) = y_0''$$

.....

$$p_{n-1}(x_0) = y_0^{(n-1)}$$

Ko'pgina hollarda umumiy yechimi analitik ko'rinishda ifodalanmagan tenglamalar bilan ish yuritishga to'g'ri keladi. Shuning uchun masalaning taqribiy yechimini beruvchi u yoki bu usullarni qo'llashga ehtiyoj seziladi. Biz eng oddiy usullardan biri – Eyler usulini ko'rib chiqamiz.

Eyler usuli. Eyler usuli ikki tartibli tenglamalarga nisbatan qo'llab ko'raylik. Aytilganlarni hech qiyinchiliksiz yuqori darajali tenglama hollariga ham umumlashtirishimiz mumkin.

Ikkinchi tartibli tenglama berilgan bo'lsin

$$y'' = f(x, y, y')$$

bunda $y(x_0) = y_0$ va $y'(x_0) = y_0'$ boshlang'ich shartlar. Bu tenglama quyidagi sistemaga keltirilishi mumkin:

$$y' = p,$$

$$p' = f(x, y, p)$$

jumladan

$$y(x_0) = y_0,$$

$$p(x_0) = p_0 = y_0'$$

$y'' = f(x, y, y')$ tenglamaning sonli yechimini topishda masala quyidagi ko'rinishda qo'yiladi: $x_0, x_1, \dots, x_n$ nuqtalarda $y(x_i), i = 0, 1, \dots, n$ ning aniq yechimini toppish uchun $y_0, y_1, \dots, y_n$ yaqinlashishlar topilsin.

$\Delta x_i = x_{i+1} - x_i$ ayirmani $h_i, i = 0, 1, \dots, n-1$ orqali ifodalaymiz. Ko'pgina hollarda h_i kattalikni doimiy deb olish qulay, shunda

$$x_i = x_0 + ih, i = 0, 1, \dots, n$$

Hosilani ta'rifi bo'yicha quyidagiga ega bo'lamiz

$$\frac{dy}{dx} = \lim_{h \rightarrow 0} \frac{y(x+h) - y(x)}{h},$$

$$\frac{dp}{dx} = \lim_{h \rightarrow 0} \frac{p(x+h) - p(x)}{h}$$

yoki

$$y(x+h) = y(x) + y'h + O_1(h),$$

$$p(x+h) = p(x) + p'h + O_2(h)$$

bunda y' va p' x nuqtada olinadi hamda h nolga intilganda $O_1(h)$ va $O_2(h)$ nolga intiladi.

$$y(x+h) = y(x) + y'h + O_1(h),$$

$$p(x+h) = p(x) + p'h + O_2(h) \quad \text{da} \quad x \quad \text{ning} \quad o'rniga$$

$x_i, i = 0, 1, \dots, n-1$ qiymatlarni hamda y' va p' o'rniga ularning $y' = p$,

$p' = f(x, y, p)$ dagi ifodasini qo'yib quyidagiga ega bo'lamiz:

$$y(x_{i+1}) = y(x_i) + hp(x_i) + O_1(h),$$

$$p(x_{i+1}) = p(x_i) + hf(x_i, y(x_i), p(x_i)) + O_2(h)$$

Bundan ko'rinib turibdiki, y va p funksiyalarning x_{i+1} nuqtadagi taqribiy qiymatini quyidagi usul bilan hisoblash maqsadga muvofiq bo'ladi:

$$y_{i+1} = y_i + hp_i,$$

$$p_{i+1} = p_i + hf(x_i, y_i, p_i)$$

bunda

$$\begin{aligned}y_{i+1} &= y(x_{i+1}), p_{i+1} = p(x_{i+1}), \\ y_i &= y(x_i), p_i = p(x_i), i = 0, 1, \dots, n-1\end{aligned}$$

Misollar

292. $[0;2]$ oraliqda quyidagi differensial tenglamaning taqribiy yechim qiymatlari jadvali tuzilsin

$$y'' - 3y' + 2y - 2x + 3 = 0$$

Bunda $y(0) = 1, y'(0) = 2$ shart qanoatlantirilsin, integrallashtirish qadami qilib $h=0,2$ tanlansin. Berilgan tenglamaning taqribiy yechimini Eyler usuli yordamida topilsin. X ning qiymatini va $y = e^x + x$ aniq yechimini chop etishga alohida e'tibor berilsin. Hisoblash natijalarini massiv ko'rinishida tasvirlansin.

Berilgan massivda 3 ta massiv chop etilishi kerak: birinchisi- x ning 0,2 qadam bilan 0 dan 2 gacha bo'lgan qiymati (dasturda bu massiv X orqali ifodalangan), ikkinchisi yaqinlashtirilgan yechim qiymati (dasturda bu massiv Y orqali ifodalangan), uchinchisi- aniq yechim qiymat (dasturda bu massiv Z orqali ifodalangan). X ning 0,2 qadam bilan 0 dan 2 gacha o'zgarganda biz x ning 11 ta qiymatini hosil qilamiz. X ning har bir qiymatiga y va z ning qiymatlari mos keladi, shuning uchun 3 ta massiv ham 11 ta komponentga ega bo'ladi. Dasturda birinchi hosila qiymati uchun 11 o'lchovli yana bir massiv hisobga olingan (uni P deb ataymiz).

$$\begin{aligned}y'' - 3y' + 2y - 2x + 3 &= 0 & \text{tenglamani} & \begin{aligned}y' &= p, \\ p' &= f(x, y, p)\end{aligned}\end{aligned}$$

ko'rinishiga quyidagi usul bilan keltiraylik:

$$\begin{aligned}y' &= p, \\ p' &= 3p - 2y + 2x - 3, \\ y(0) &= 1, \\ p(0) &= 2\end{aligned}$$

Quyida keltirilgan algoritm asosida tuzilgan dastur keltirilgan.

```

#include <iostream>
using namespace std;
int main() {
float x[100],y[100],h=0.2;
int i,n=10;
x[0]=0;
y[0]=2;
for(i=0;i<11;i++) {
x[i+1]=x[i]+h*i;
y[i+1]=y[i]+h*3*y[i]-2*y[i]+2*x[i]-3;
cout<<x[i]<<"\t"<<y[i]<<endl;
}
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\3.exe
0      2
0      -3.8
0.2    -1.48
0.6    -2.008
1.2    -0.9968
2      -0.20128
3      1.08051
4.2    2.5678
5.6    4.37288
7.2    6.45085
9      8.81966
Для продолжения нажмите любую клавишу . . .

```

Mustaqil ishlash uchun vazifa

293. Eyler usuli bilan h qadam bilan $[a, b]$ oraliqda quyida keltirilgan jadvalda ko‘rsatilgan differensial tenglamalarning yechim natijalaridan iborat jadvalni tuzish maqsadida hisoblash formulalari yozib olinsin. Dasturda x argumentning qiymati va aniq echimning chop etilishi hisobga olinsin. Hisoblash natijalari massivda ifodalansin.

	Differensial tenglama	Boshlan-g'ich shartlar	Integ-rallash tirish orali-g'i	Integra llashti-rish qadami	Aniq yechim
.	$(1+x^2)y'' + (y')^2 + 1 = 0$	$y(0)=1$ $y'(0)=1$	$[0;0,5]$	0,05	$1-x+2\ln(1+x)$

.	$y'' + 2y' + 2y = 2e^{-x} \cos x$	$y(0)=1$ $y'(0)=1$	[0;0,5]	0,05	$e^{-x}(\cos x + \sin x + x \sin x)$
.	$y'' + 4y = e^{3x}(13x - 7)$	$y(0)=0$ $y'(0)=-4$	[0;0,2]	0,02	$\cos 2x - \sin 2x + e^{3x}(x - 1)$
.	$y'' 4y' + 4y = 0$	$y(0)=1$ $y'(0)=-1$	[0;1]	0,1	$(1+x)e^{-2x}$
.	$y'' - y = \sin x + \cos 2x$	$y(0)=1,8$ $y'(0)=-0,5$	[0;2]	0,2	$e^x + e^{-x} - \frac{1}{2}\sin x - \frac{1}{5}\cos 2x$
.	$y'' - 3y' = e^{5x}$	$y(0)=2,2$ $y'(0)=0,8$	[0;0,2]	0,02	$2 + 0,1(e^{3x} + e^{5x})$
.	$y'' + 4y = \cos 3x$	$y(0)=0,8$ $y'(0)=2$	[0;1]	0,1	$\cos 2x + \sin 2x - 0,2 \cos 3x$
.	$y'' - y' - 6y = 2e^{4x}$	$y(0)=1,4$ $y'(0)=0,4$	[0;1]	0,1	$0,1e^{3x} + e^{-2x} + \frac{1}{3}e^{4x}$
.	$y'' - 2y' + y = 5xe^x$	$y(0)=1$ $y'(0)=2$	[0;1]	0,1	$e^x + xe^x + 5e^x \frac{x^3}{6}$
0	$y'' + y' - 6y = 3x^2 - x - 1$	$y(0)=-0,9$ $y'(0)=3,2$	[0;1]	0,1	$0,1e^{2x} - e^{-3x} 0,5x^2$
1	$8y'' + 2y' - 3y = x + 5$	$y(0)=1/9$ $y'(0)=-7/12$	[0;1]	0,1	$e^{\frac{x}{2}} + e^{\frac{3x}{4}} - \frac{x}{3} - \frac{17}{9}$
2	$x^2 y'' - 2y = 0$	$y(1)=5/6$ $y'(1)=2/3$	[1;2]	0,1	$\frac{1}{2}x^2 + \frac{1}{3x}$
3	$y'' - 4y' + 5y = 3x$	$y(0)=1,48$ $y'(0)=3,6$	[0;0,5]	0,05	$e^{2x}(\cos x + \sin x) + \frac{3}{5}x + \frac{12}{25}$
4	$y'' - 5y' + 6y = e^x$	$y(0)=0$ $y'(0)=0$	[0;0,2]	0,02	$-e^{2x} + 0,5e^{3x} + 0,5e^x$
5	$y'' - 3y' + 2y = x^2 + 3x$	$y(0)=5,1$ $y'(0)=4,2$	[0;1]	0,1	$e^x + 0,1e^{2x} + \frac{x^2}{2} + 3x + 4$
6	$y'' + y = 1 + e^x$	$y(0)=2,5$ $y'(0)=1,5$	[0;1]	0,1	$\cos x + \sin x + 1 + \frac{1}{2}e^x$
7	$y'' + \frac{1}{x}y' - \frac{1}{x^2}y = 8x$	$y(1)=4$ $y'(1)=4$	[1;1,5]	0,05	$2x + \frac{1}{x} + x^3$

8	$x^2 y'' + xy' = 0$	$y(1)=5$ $y'(1)=-1$	[1;1,5]	0,05	$5 - \ln x$
9	$y'' - 2y' + y = xe^x$	$y(0)=1$ $y'(0)=20$	[0;0,5]	0,05	$(1+x)e^x + x^3 e^x / 6$
0	$x^2 y'' + 2,5y' - y = 0$	$y(1)=2$ $y'(1)=3,5$	[1;2]	0,1	$3\sqrt{x} - x^{-2}$
1	$4xy'' + 2y' + y = 0$	$y(1)=1,4$ $y'(1)=-0,2$	[1;2]	0,1	$\sin\sqrt{x} + \cos\sqrt{x}$
2	$x^2 y'' - 4xy' + 6y = 2$	$y(1)=1,4$ $y'(1)=2,3$	[1;2]	0,1	$x^2 + 0,1x^3 + \frac{1}{3}$
3	$y'' - y = e^{2x}(x-1)$	$y(0)=11/9$ $y'(0)=-\frac{11}{9}$	[0;1]	0,1	$e^x + e^{-x} + e^{2x}(\frac{1}{3}x - \frac{7}{9})$
4	$y'' - 0,5y' - 0,5y = 3e^{x/2}$	$y(0)=-4$ $y'(0)=-2,5$	[0;1]	0,1	$e^x + e^{-x/2} - 6e^{x/2}$
5	$y'' - 4y' + 4y = 2x - 3$	$y(0)=-1/4$ $y'(0)=-1/2$	[0;0,05]	0,05	$(1+x)e^{-2x} + \frac{x}{2} - \frac{5}{4}$
6	$y'' + y = x^2 - x + 2$	$y(0)=1$ $y'(0)=0$	[0;1]	0,1	$\cos x + \sin x + x^2 - x$

KO'P O'LCHOVLI MASSIVLAR BILAN ISHLASH

Massivning o'lchov sonini (ya'ni elementning massivdagi holatini aniqlovchi indekslar soni) anglatuvchi massiv sig'imi bilan massivdagi elementlar sonini anglatuvchi massiv o'lchovi o'rtasidagi farqni bilish zarur.

C++ dasturlash tilida ko'p o'lchovli massivlar quyidagicha tavsiflanadi:

`<tur> <massiv_nomi>[<indeks1>]...[<indeksN>];`

Qancha indekslar ko'rsatilsa, massivning oldiga o'sha son qo'shib aytiladi. Ko'pincha ikki va uch o'lchovli massivlar bilan ishlaymiz. Bir o'lchovli massiv matematikadagi vektor to'shunchasiga, ikki o'lchovli massiv esa matritsa to'shunchasiga mos keladi.

Misollar

294. Chiziqli tenglamalar sistemasini Gauss usuli bilan yechish dasturini tuzing.

Bu usul 2 ta bosqichga ajraladi. Birinchi bosqichda chiziqli sistema matritsasi uchburchakli ko'rinishga keltiriladi, ikkinchi bosqichda esa uchburchakli chiziqli sistema yechiladi.

Boshlang'ich sistemadan uchburchakli sistemaga o'tish uchun, asosiy dioganal ostidagi barcha elementlarni quyidagi usulda 0 ga tenglash zarur.

1) $i=1$ o'rnatamiz

2) agar $a_{ii} \neq 0$, i va $i+1, i+2, \dots, n$ qatorlarni chiziqli kombinatsiyalash yordamida $a_{i+1i}, a_{i+2i}, \dots, a_{ni}$ elementlarni olib tashlash mumkin; so'ngra $i=i+1$ o'rnatiladi;

3) $a_{ii} = 0$ bo'lgan holda, keyingi qatorlar ichida $a_{hi} \neq 0$ elementlar topiladi va i hamda h qatorlarning o'rnini almashtirilib, so'ngra 2 ga qaytiladi;

4) Agar $a_{nn} = 0$ bo'lsa, unda matritsaning determinanti 0 ga teng bo'ladi va yechish jarayoni to'xtatiladi; agar $a_{nn} \neq 0$ bo'lsa, unda uchburchak matritsaga keltirish jarayoni to'xtatiladi va keyingi bosqichga o'tish imkoniyati tug'iladi.

Quyida keltirilgan algoritm asosida tuzilgan dastur ko'rsatilgan:

```

#include <cstdlib>
#include <iostream>
using namespace std;
int main(int argc, char *argv[]) {
    int j,i,n;
    float a[100][100];
    cout<<"n->";
    cin>>n;
    cout<<"Tenglama koefitsentlarini kiriting:\n";
    for(i=1;i<=n;i++)
    for(j=1;j<=n;j++) {
        cout<<"a["<<i<<"["<<j<<"]: ";
        cin>>a[i][j]; }
    if(a[1][1]!=0)
    for(i=1;i<=n;i++)
        a[i][i+1]=0;
    for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
        cout<<"a["<<i<<"["<<j<<": "<<a[i][j]<<endl;
    return 0;}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\4.exe
n->2
Tenglama koefitsentlarini kiriting:
a[1][1]: 6
a[1][2]: 7
a[2][1]: 5
a[2][2]: 4
a[1][1]: 6
a[1][2]: 0
a[2][1]: 5
a[2][2]: 4
Для продолжения нажмите любую клавишу . . .

```

295. Beshta tsexning har 4 ta bo‘limida xom ashyo miqdori haqida ma’lumotnoma jadval ko‘rinishida ifodalangan. Eng kam xom ashyo mavjud bo‘lgan sexning nomerini aniqlang.

```

#include <iostream>
using namespace std;
int main(){
    float a[4][5];
    int k,i,j;
    float mina;
    cout<<"Massiv elementlarini kiriting";

```

```

for(i=1;i<=4;i++)
for(j=1;j<=5;j++)
cin>>a[i][j];
for(i=1;i<=4;i++)
for(j=1;j<=5;j++){
if(a[i][j]<mina)
mina=a[i][j];
k=j;
}
cout<<"mina="<<mina<<"k="<<k;
return 0;
}

```

```

D:\Fanlar\OYDT\Дастурлар\азик\6.exe
Massiv elementlarini kiriting4
5
6
7
5
3
4
7
8
9
8
7
6
5
4
3
22
3
4
5
mina=2.67416k=5Для продолжения нажмите любую клавишу . . .

```

296. A[1..3, 1..15] massiv berilgan. Ma'lumki uning elementlari ichida 2 tasi va faqat 2 tasi o'zaro teng. Ularning indekslari chop etilsin.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int a[3][15];
int i,j,k,l;
cout<<"Massiv elementlarini kiriting";
for(i=1;1<3;i++)
for(j=1;j<15;j++)
cin>>a[i][j];
for(i=1;1<3;i++)

```

```

for(j=1;j<15;j++)
for(k=i;k<3;k++)
for(l=j;l<15;l++)
if(a[i][j]=a[k][l])
cout<<"i="<<i<<"j="<<j;
return 0;
}

```

```

C:\Program Files\Borland\CBUILDER6\Projects\Project1.exe
Massiv elementlarini kiriting4
3
4
5
6
7
8
9
8
7
66

```

297. a_1, a_2, a_3 butun sonlar berilgan. $B[1..3, 1..3]$ massivni xosil qiling. Bunda $B[i, j] = a_1 - a_3$;

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
int b[3][3];
int a[3];
int i,j;
cout<<"Massiv elementlarini kiriting";
for(i=1;i<3;i++)
cin>>a[i];
for(i=1;i<3;i++)
for(j=1;j<3;j++)
b[i][j]=a[i]-3*a[i];
for(i=1;i<3;i++)
for(j=1;j<3;j++)
cout<<"b="<<b;
return 0;
}

```

```

C:\Program Files\Borland\CBUILDER6\Projects\Project1.exe
Massiv elementlarini kiriting6
5
b=1244968b=1244968b=1244968b=1244968Для продолжения нажмите любую клавишу . . .

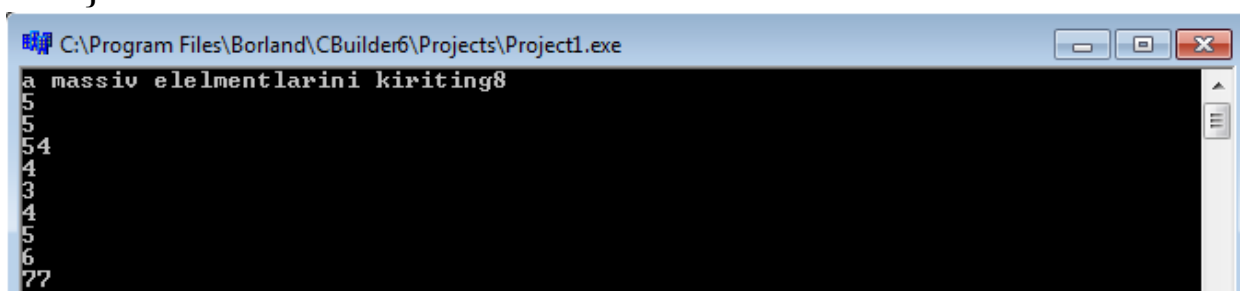
```

298. Ikkita matrisa berilgan. Ularning ko'paytmasini hosil qiling.

```

#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a[5][5];
float b[5][5];
float c[5][5];
int i,j,k;
cout<<"a massiv elementlarini kiriting";
for(i=1;i<=5;i++)
for(j=1;j<=5;j++)
cin>>a[i][j];
cout<<"b massiv elementlarini kiriting";
for(i=1;i<=5;i++)
for(j=1;j<=5;j++)
cin>>b[i][j];
for(i=1;i<=5;i++)
for(j=1;j<=5;j++)
c[i][j]=0;
for(k=1;k<=5;k++)
c[i][j]=c[i][j]+a[i][k]*b[k][j];
return 0;
}

```



299. $5 \times 5 \times 5$ kattalikka ega bo'lgan butun sonli 3 o'lchovli massivni hosil qiling va qator bo'yicha ekranga chiqaring. Eng katta elementni toping va uni massivda joylashuvini aniqlang.

```

#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float A[5][5][5];
int i,j,k,il=0,jl=0,kl=0;

```

```

float maxa;
cout<<"massiv elementlarini kiriting";
for(i=1;i<=5;i++)
for(j=1;j<=5;j++)
for(k=1;k<=5;k++)
cin>>A[i][j][k];
maxa=A[0][0][0];
for(i=1;i<5;i++)
for(j=1;j<5;j++)
for(k=1;k<5;k++)
if(A[i][j][k]>maxa){
maxa=A[i][j][k];
il=i;
jl=j;
kl=k;
}
cout << "il=" << il << "jl=" << jl << "kl=" << kl
<< "maxa=" << maxa;
return 0;
}

```

```

C:\Program Files\Borland\CBUILDER6\Projects\Project1.exe
massiv elementlarini kiriting23
43
23
43
23
44
5
6
il=1jl=1kl=1maxa=5Для продолжения нажмите любую клавишу . . .

```

Mustaqil ishlash uchun vazifa

300. Chiziqli tenglamalar sistemasini KRAMER USULI yordamida yechish dasturini tuzing.

301. $A[1...5,1...3]$ haqiqiy massiv berilgan. Uning manfiy elementlar indekslarini chop eting.

302. Ikki o'lchovli butun sonlardan iborat $A[1...10,1...7]$ massivini tuzing va ikkalasi ham toq indeksga ega bo'lgan barcha elementlar yig'indisini hisoblang.

303. $A[1...5,1...3]$ butun sondan iborat massiv berilgan. 5 ga karrali bo'lgan elementlarni va uning indekslarini ekranga chiqaring.

304. Klaviatura orqali 3×3 matrisaning butun sonli elementlarini kiriting va har bir ustunning elementlar yig'indisini hisoblang.

305. $B[1...5, 1...5]$ butun sonli massiv berilgan. Uning chap va o'ng dioganallarini chop eting.

306. $B[1...5, 1...5]$ butun sonli massiv berilgan. Uning o'ng dioganaldagi elementlar yig'indisini va chap dioganaldagi elementlar ko'paytmasini toping.

307. $B[1...5, 1...5]$ butun sonli massiv berilgan. Massivning eng kata elementi topilsin va uning chap diogonalga nisbatan joylashuvi haqida ma'lumot bering.

308. $B[1...5, 1...5]$ butun sonli massiv berilgan. Massivning eng kichik elementi topilsin va uning o'ng diogonalga nisbatan joylashuvi haqida ma'lumot bering.

309. $B[1...5, 1...5]$ butun sonli massiv berilgan. Ushbu massivning chap diogonalidan yuqorida joylashgan elementlarning yig'indisi hisoblansin.

310. $B[1..5, 1..5]$ butun sonli massiv berilgan. Ushbu massivning chap diagonalidan pastga joylashgan elementlarning ko'paytmasi hisoblansin.

311. $B[1..5, 1..5]$ butun sonli massiv berilgan. Ushbu massivning chap diagonalidan pastda joylashgan manfiy sonlar miqdorini aniqlang.

312. $B[1..5, 1..5]$ butun sonli massiv berilgan. Ushbu massivning chap diagonalidan yuqorida joylashgan musbat sonlar miqdorini aniqlang.

313. $a_1, a_2, \dots, a_{10}, \quad b_1, b_2, \dots, b_{20}$ haqiqiy sonlar berilgan. $C[1..20, ..10]$ haqiqiy massivni hosil qiling, bunda $C[i, j] = a_i(1 + |b_j|)$.

314. $A[1..10, 1..12]$ butun sonli massivni hosil qiling, bunda $a = i + 2j$.

315. 5×5 matritsani hosil qiling bunda har bir element qiymati u joylashgan qator va ustun kesishmasining yig'indisiga teng va har bir qatordagi elementlar yig'indisini hisoblang.

316. $A[1..7, 1..7]$ haqiqiy massivni hosil qiling. Uning birinchi qatori $a = 2j + 3$ formulasi bilan, 2 chi qator $a_{2j} = j + 3 / (2 + j)$ formulasi bilan berilmoqda. Keyingi har qaysi qator oldingi ikkita qator yig'indisidir.

317. n natural soni berilgan. Agar $a_{ij} = \sin(i+j / 2)$ bo'lsa $A[1..n, 1..n]$ massivida nechta musbat element borligini aniqlang.

318. $A[1..4, 1..5]$ butun sonli massiv berilgan. Har bir ustunning o'rta arifmetigi hisoblansin.

319. $n*m$ kattalikli haqiqiy massiv berilgan, unda hamma elementlar ham 0 ga teng emas. Berilgan massivdagi elementlarni o'zining modul bo'yicha eng katta elementiga bo'lish yo'li orqali yangi massivni hosil qiling.

320. $A[1..4, 1..5]$ butun sonli massiv berilgan. Matritsani o'zgartirish talab etilmoqda; elementma-element oxirgi qatorni oxirgi qatordan tashqari barcha qatorlardan ayrilib chiqilsin.

321. C ikki o'lchovli massivda har bir ustun elementlarini o'sib boruvchi tartibida joylashtiring.

322. $m*n$ kattalikga ega bo'lgan matritsada har bir qatorda elementlarni kamayib borish tartibida joylashtiring.

323. $A[1..4, 1..5]$ butun sonli massiv berilgan. Massivning shunday qator nomerlarini toping-ki ularda 10 ga teng bo'lgan aqalli bitta element mavjud bo'lsin.

324. $m*n$ kattalikga ega bo'lgan matritsaning har bir ustun elementlarini o'sib borish tartibida joylashtiruvchi dasturni tuzing.

325. $A[1..5, 1..5]$ massiv berilgan. Yangi massivni shunday yo'l bilan hosil qiling-ki, birinchi massivda har bir qator elementlarini chap diagonal bo'yicha turuvchi ushbu qator elementlariga bo'lish jarayoni amalga oshsin.

327. $A[1..5, 1..5]$ massiv berilgan. Unda birinchi va oxirgi qator o'rnilari almashtirilsin.

328. $A[1..5, 1..6]$ massiv berilgan. Undagi birinchi va oxirgi ustun o'rnilari almashtirilsin.

329. To'g'ri burchakli matritsa berilgan. Elementlar yig'indisi eng katta bo'lgan qatorni toping.

330. To'g'ri burchakli matritsa berilgan. Elementlar yig'indisi eng katta bo'lgan ustunni toping.

331. $4*8$ kattalikka ega bo'lgan butun sonli 2 o'lchovli massivda barcha juft sonlar yig'indisi topilsin.

332. $5*5$ kattalikka ega bo'lgan butun sonli 2 o'lchovli massivda asosiy diagonalda joylashgan barcha elementlar yig'indisi topilsin.

333. 7×4 kattalikka ega bo'lgan butun sonli 2 o'lchovli massivda eng katta elementning qator va ustun nomeri aniqlansin.

334. 6×5 kattalikka ega bo'lgan butun sonli 2 o'lchovli massiv berilgan. Elementlarining o'rta arifmetik qiymati eng katta bo'lgan qator no'meri aniqlansin.

335. 5×9 kattalikka ega bo'lgan butun sonli 2 o'lchovli massivda bir xil nomerli qator va ustun o'rinlari almashtirilsin.

336. Butun sonli ikki o'lchovli massivda qatorning eng katta qiymatlari orasida eng kichigi aniqlansin. Shunday elementning qator va ustun nomeri aniqlansin.

337. Butun sonli ikki o'lchovli massivda ushbu massiv uchun eng katta elementi joylashgan ustunni olib tashlang.

338. Butun sonli ikki o'lchovli massivda barcha takrorlanmaydigan elementlar topilsin.

339. Ikki o'lchovli massivni spiral bo'yicha 1 dan to 100 gacha bo'lgan butun sonlar bilan to'ldiring.

340. Butun sonli ikki o'lchovli massivda birinchi qator elementlari yig'indisidan kichik bo'lgan barcha elementlarni ushbu yig'indiga almashtiring.

341. Butun sonli massiv qatorlarini kamayib borish tartibini joylashtiring.

342. Massivning toq ustunlarini o'sib borish tartibida joylashtiring.

343. Ustaxona tomonidan ishlab chiqariladigan turli xil uskunalarning narxi va nomlari berilgan. Ularni narxi va alfabit bo'yicha tartiblang.

344. O'quvchilarning familiyasi va telefon nomerlarini o'zida mujassamlashtirgan ikki o'lchovli massiv berilgan. O'quvchining familiyasi bo'yicha uning telefon nomerini aniqlang.

345. Ikki o'lchovli massiv elementlari sehrli kvadratni hosil qilishini tekshiruvchi dasturni tuzing (sehrli kvadratda sonlar yig'indisi barcha vertikal, gorizontallar va ikkita diagonal bo'yicha bir hil).

346. Matrisaning bosh elementlarini diagonalga nisbatan simmetrik ravishda o'rinlarini almashtiruvchi dasturni tuzing.

347. Ikki o'lchovli K massiv ustunlarini davriy o'rin almashtirish jarayonini amalga oshiruvchi dasturni tuzing, bunda i -ustun $i+1$ -oxirgisi esa birinchi bo'lib qoladi.

348. Ikki o'lchovli A massivda no'llik elementlariga ega bo'lmagan qatorlarda turuvchi barcha musbat sonlarning yigindisini aniqlang.

349. Kvadratli jadvalda kesishmasida musbat elementlar ichida eng kichigi joylashgan qator va ustun elementlar o'rnolari almashtirilsin.

350. Butun sonli 2 o'lchovli massivni tuzing. Kesishmasida eng kichik element joylashgan qator va ustunni olib tashlang.

351. To'g'ri burchakli jadvalda 2-dan boshlab har bir qatorning eng kichik elementini oldingi qatordagi eng katta element bilan almashtiring.

352. Uchta jadval berilgan. 2 ta A va B chiziqli va to'g'ri burchakli C jadvallar. A chiziqli jadvalda 11 sinf o'quvchilarining familiyalari yozilgan, to'g'ri burchaklisida esa ularning birinchi yarim yillik uchun 10 ta fan bo'yicha olgan baholari kiritilgan, chiziqli B jadvalda ushbu fanlarning nomlari yozilgan. O'zlashtira olmagan o'quvchilar ro'yxatini tuzing. C jadvaldan 3 tadan ortiq 2 olgan o'quvchining barcha baholari, A ro'yxatdan esa o'quvchining familiyasi olib tashlansin. Qaysi fan bo'yicha eng katta o'rtacha ball kuzatilganini aniqlang. Barcha fanlar bo'yicha eng katta o'rtacha ball to'plagan ixtiyoriy 3 ta o'quvchini toping. O'quvchilarni ularning o'rtacha ballari kamayib borishi tartibida ro'yxatni tuzing. O'quvchilar ro'yxatini alifbo bo'yicha tartiblangan deb hisoblab, unga "Abdullayev Zoir 543542535" qatorni kiriting.

353. 10×12 kattalikka ega bo'lgan butun sonlar matritsasi berilgan. Uning barcha ulanish nuqtalarining indekslarini chop eting (Ulanish nuqtasi deb, o'zing qatorida eng kichik bo'lib, o'zining ustunida eng katta bo'lgan va ustinida eng katta bo'lgan, va aksincha o'zining qatorida eng katta va ustunida eng kichik bo'lgan elimetga aytiladi).

MASSIVLAR USTIDA AMALLAR BAJARISH

Berilgan ish tarkibida – ichida takrorlanishlar mavjud bo‘lishi mumkin bo‘lgan sikllarga (ko‘pincha ikkilik) olib kelinuvchi masalalar hamda ularning yechim algoritmlari bor.

Tuzilgan dastur umumiy ko‘rinishda bo‘lishi kerak, ya’ni qo‘yilgan paytda kiritilgan ma’lumotlar o‘zgarganda dastur o‘z holatini almashtirishi kerak emas.

Barcha masalalarda ko‘pincha massivlardan foydalanish talab etiladi, jumladan massiv kattaligi masala tuzish jarayonida ko‘rsatilmagan bo‘lib, “Kiruvchi ma’lumotlar” sifatida beriladi. Bunda dasturni shunday tuzish kerakki, u ixtiyoriy kattalikdagi (tabiiyki, dasturda ko‘rsatilgan cheklanishlar me’yorida) massivlar uchun qo‘llanishga loyiq bo‘lishi kerak. Buning uchun dasturda masala tuzish jarayonidagi cheklanishlardan kelib chiqqan holda maksimal kattalikdagi massivni (yoki massivlarni) ta’riflash kerak.

Misollar

354. Berilgan $A = \{a_{ij}\}$ matritsaning kvadrati bo‘lgan $B = \{b_{ij}\}$ matritsa elementlarini hisoblovchi dasturni tuzing. B matritsaning har bir komponenti quyidagi formula asosida hisoblanadi:

$$b_{ij} = \sum_{k=1}^n a_{ik} a_{kj}, j = 1, 2, \dots, n.$$

$B = \{b_{ij}\}$ matritsadan $y = \{y_1, y_2, \dots, y_n\}$ vector olinsin, bunda har bir y_j komponenta B matritsaning j- chi ustundagi elementlar yig‘indisi sifatida olinadi, ya’ni quyidagi formula asosida hisoblanadi.

$$y_j = \sum_{i=1}^n b_{ij}, j = 1, 2, \dots, n.$$

$n(n \leq 15)$ ning qiymati va A matritsa elementlari berilgan.

Ushbu algoritmgaga quyidagi dastur to‘g‘ri keladi.

```
#include <iostream>
```

```

using namespace std;
int main(int argc, char *argv[]){
float a[15][15];
float b[15][15];
float y[15];
int N,i,j,k;
cout<<"Massiv kattaligini kiriting";
cin>>N;
cout<<"a massiv elementlarini kiriting";
for(i=1;i<N;i++)
for(j=1;j<N;j++)
cin>>a[i][j];
for(i=1;i<N;i++)
for(j=1;j<N;j++){
b[i][j]=0;
for(k=1;k<N;k++)
b[i][j]=b[i][j]+a[i][k]*a[k][j];
}
for(i=1;i<N;i++)
for(j=1;j<N;j++)
cout<<b[i][j];
return 0;
}

```

355. $x = x_i$ argument qiymati uchun chiziqli interpozitsiya formulasi bo'yicha $y(x)$ funksiyani hisoblovchi dasturni tuzing.

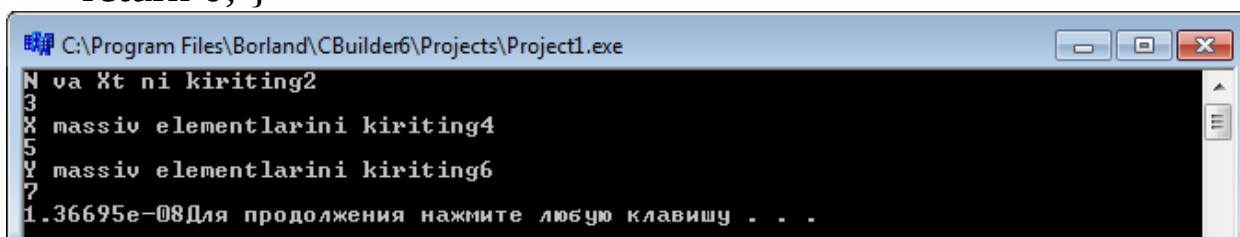
$$y(x_t) = y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i} (x_t - x_i),$$

x_t ikkita jadval ko‘rinishida berilgan x_i qiymat o‘rtasida ifodalangan, ya’ni $x_i \leq x_t \leq x_{i+1}$, $1 \leq i \leq n-1$

$n, x_1, x_2, \dots, x_n; y_1, y_2, \dots, y_n$ ($n \leq 15$) qiymatlar berilgan.

Quyidagi dasturlar keltirilgan:

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float x[15];
float y[15];
int N,i;
float xt,yxt;
cout<<"N va Xt ni kiriting";
cin>>N>>xt;
cout<<"X massiv elementlarini kiriting";
for(i=1;i<=N;i++)
cin>>x[i];
cout<<"Y massiv elementlarini kiriting";
for(i=1;i<=N;i++)
cin>>y[i];
for(i=2;i<=N;i++)
if((x[i]<=xt)&&(xt<=x[i+1]))
yxt=y[i]+(y[i+1]-y[i])/(x[i+1]-x[i])*(xt-x[i]);
cout<<yxt;
return 0; }
```



356. t argumentning berilgan qiymatida $u_1(t) = 1$; $u_2(t) = t$ bo‘lsa, qolgan funksiyalar esa $u_{k+1}(t) = 2tu_k(t) - u_{k-1}(t)$, $k = 2, 3, \dots, n-1$. Rekurent munosabat orqali topilsa, $u_1(t), u_2(t), \dots, u_n(t)$ funksiya (Chebishev polinomi)ni hisoblovchi dastur tuzing.

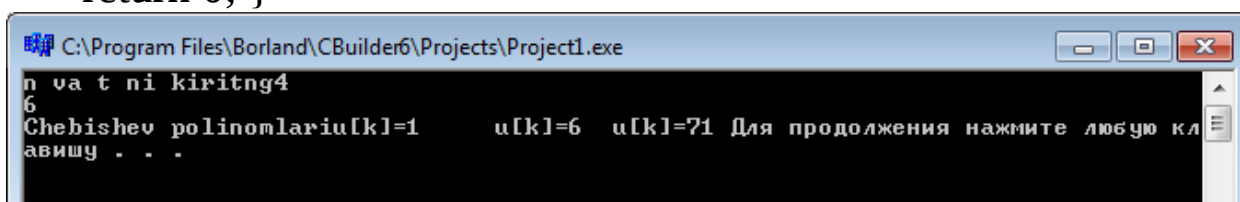
$n(n \leq 18)$ va t qiymatlari berilgan.

Ushbu algoritm asosida tuzilgan dastur quyida keltirilgan.

```

#include <iostream>
using namespace std;
int main(){
float u[18];
int n,n1,k;
float t;
cout<<"n va t ni kiritng";
cin>>n>>t;
u[1]=1;
u[2]=t;
n1=n-1;
for(k=2;k<n1;k++)
u[k+1]=2*t*u[k]-u[k-1];
cout<<"Chebishev polinomlari";
for(k=1;k<n;k++)
cout<<"u[k]="<<u[k]<<"\t";
return 0; }

```



57. Quyidagi polinomni hisoblovchi dasturni tuzing:

$$P(x) = a_1 x^n + a_2 x^{n-1} + \dots + a_{1n} x + a_{1+1},$$

$N(n \leq 12)$; $a_1, a_2, \dots, a_{n+1}, x$ qiymatlari berilgan.

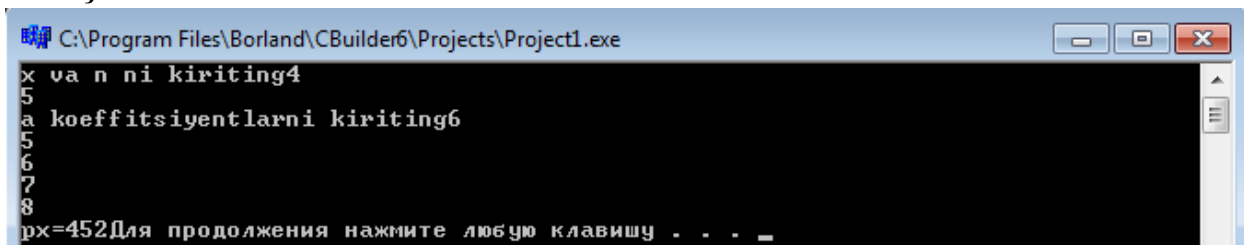
Polinomni hisoblash uchun gorner sxemasidan foydalaniladi, ya'ni $P(x)$ polinomi quyidagi ko'rinishda tasvirlanadi.

$$P(x) = (\dots ((a_1 x + a_2) x + a_3) + \dots + a_n) x + a_{n+1}.$$

Yozishning bunday shaklidan kelib chiqadi, $P_1 = a_1$ deb hisoblab va $P_i = P_{i-1} x + a_i, i = 2, 3, \dots, n + 1$, Formulani ko'p marotaba qo'llab, P_{n+1} sifatida polinomning qidirilayotgan natijasini topamiz. E'tibor berish joizki, dasturni tuzayotganda, P_i indekslashtirilgan o'zgaruvchini qo'llashga ehtiyoj tug'lmaydi, chunki P_i ning qiymati ishlatiladi, shuning uchun biz dasturda bitta oddiy P o'zgaruvchini (Bizning dasturimizda PX) qo'llasak ham bo'ladi, u hisoblash jarayonida ko'p marotaba o'z qiymatini o'zgartiradi.

Dasturi quyidagi ko‘rinishga ega:

```
#include <iostream>
using namespace std;
int main(int argc, char *argv[]){
float a[13];
int n,n1,i;
float x,px;
cout<<"x va n ni kiriting";
cin>>x>>n;
n1=n+1;
cout<<"a koefitsiyentlarni kiriting";
for(i=1;i<n1;i++)
cin>>a[i];
px=a[i];
for(i=2;i<n1;i++)
px=px*x+a[i];
cout<<"px="<<px;
return 0;
}
```



358. Ikkita $x=\{x_1,\dots,x_{10}\}$ va $y=\{y_1,\dots,y_{10}\}$ vektorlarning yig‘indisiga teng bo‘lgan $z=\{z_1,\dots,z_{10}\}$ vektorni $z_i = x_i + y_i$, $i=1,\dots,10$ formula asosida hisoblovchi dasturni tuzing.

$x = \{x_1, \dots, x_{10}\}$ vektor berilgan. Vektor komponentlari qiymati quyidagi formula asosida hisoblanadi:

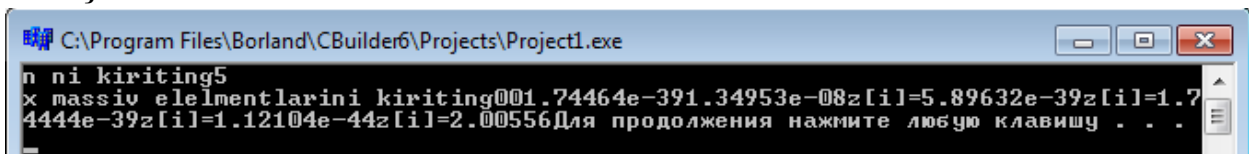
$y_i = 0,7 \sin 0,5i + 1,3\cos(i+1)$, $i=1,\dots,10$.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
float x[10];
float y[10];
float z[10];
```

```

int n,i;
cout<<"n ni kiriting";
cin>>n;
cout<<"x massiv elementlarini kiriting";
for(i=1;i<n;i++)
cout<<x[i];
for(i=1;i<n;i++)
y[i]=0.7*sin(0.5*i)+1.3*cos(i+1);
z[i]=x[i]+y[i];
for(i=1;i<n;i++)
cout<<"z[i]="<<z[i];
return 0;
}

```



359. Teskari matrisani topish dasturini tuzing.

Berilgan A matrisaga nisbatan teskari matrisa, faqat uning berilgan matrisaga ko'paytmasi o'ng tomondan ham, chap tomondan ham birlik E matrisani bersagina bo'ladi.

Buni quyidagi ko'rinishida yozib olish mumkin:

$$A * A^{-1} = A^{-1} * A = E,$$

bu yerda A^{-1} - teskari matrisa.

A^{-1} teskari matrisani topish chiziqli tenglamalar sistemasini yechishda muhim ahamiyatga ega.

```

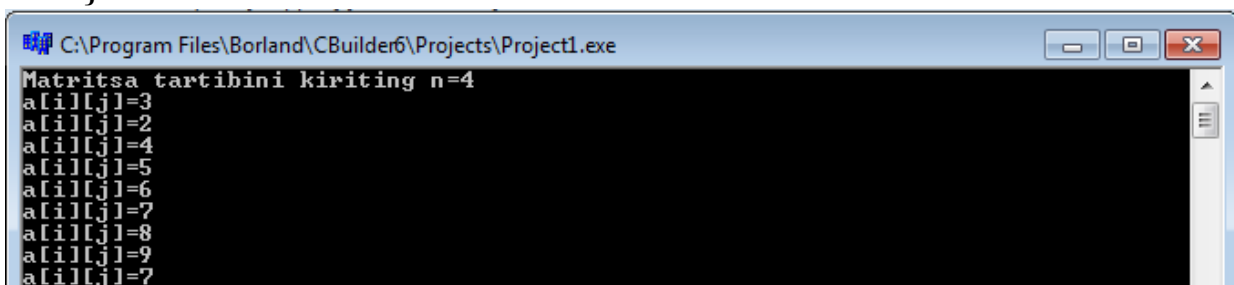
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int k,m,n,i,l,j;
float a[10][10];
float b[10][10];
float c[10][10];
float d;
cout<<"Matritsa tartibini kiriting n=";
cin>>n;
for(i=1;i<n;i++)

```

```

for(j=1;j<n;j++){
cout<<"a[i][j]=";
cin>>a[i][j];
}
for(i=1;i<n;i++)
for(j=1;j<n;j++)
if(i=j)
b[i][j]=1;
else b[i][j]=0;
for(i=1;i<n-1;i++)
for(j=i+1;j<n;j++)
d=a[j][i]/a[i][i];
for(l=i+1;l<n;l++)
a[j][l]=a[j][l]-d*a[i][l];
for(k=1;k<n;k++)
b[j][k]=b[j][k]-d*b[i][k];
for(i=1;i<n;i++)
c[n][i]=b[n][i]/a[n][n];
for(l=n-1;l<n;l--)
d=0;
for(j=l+1;j<n;j++)
d=d+a[l][j]*c[j][i];
c[l][i]=(b[l][i]-d)/a[l][l];
cout<<"Teskari matritsa";
for(i=1;i<n;i++)
for(j=1;j<n;j++)
cout<<c[i][j];
return 0;
}

```



The screenshot shows a window titled "C:\Program Files\Borland\CBUILDER6\Projects\Project1.exe". The output text is as follows:

```

Matritsa tartibini kiriting n=4
a[i][j]=3
a[i][j]=2
a[i][j]=4
a[i][j]=5
a[i][j]=6
a[i][j]=7
a[i][j]=8
a[i][j]=9
a[i][j]=7

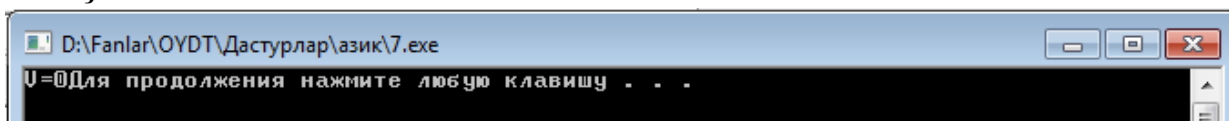
```

360. Uchlari $P_1(x_1, y_1, z_1)$, $P_2(x_2, y_2, z_2)$, $P_3(x_3, y_3, z_3)$, $P_4(x_4, y_4, z_4)$, bo‘lgan tetraedr hajmini quyidagi formula asosida hisoblang:

$$V = \frac{1}{6} \begin{vmatrix} x_1 & y_1 & z_1 & 1 \\ x_2 & y_2 & z_2 & 1 \\ x_3 & y_3 & z_3 & 1 \\ x_4 & y_4 & z_4 & 1 \end{vmatrix}$$

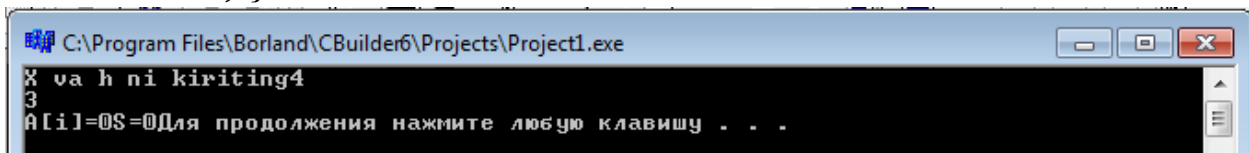
Agar determinant manfiy bo‘lsa, hajm V sifatida uning absolyut kattaligi olinadi.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int n,i,j,k,l,m;
float d,V;
float a[10][10];
float x1,x2,x3,x4,y1,y2,y3,y4,z1,z2,z3,z4;
n=4;
a[1][1]; a[1][2]; a[1][4]; a[1][3];
a[2][1]; a[2][2]; a[2][4]; a[2][3];
a[3][1]; a[3][2]; a[3][4]; a[3][3];
a[4][1]; a[4][2]; a[4][4]; a[4][3];
for(k=1;k<n-1;k++)
for(m=k+1;m<n;m++)
for(l=k+1;l<n;l++)
a[m][l]=a[m][l]+(a[m][k]*a[k][l])/a[k][k];
d=1;
for(k=1;k<n;k++)
d=d*a[k][k];
V=d/6;
cout<<"V="<<V;
return 0;
}
```



361. $a=\{a_1,...,a_{21}\}$ vektorni $\cos x, \cos(x+h),..., \cos(x+20h)$ ketma-ketlik hadlaridan hosil qiling. Ketma-ketlikda absolyut kattaligi bo'yicha 0,5 dan katta bo'lgan hadlarning yig'indisi topilsin.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
int i;
float X,h,S;
float A[21];
cout<<"X va h ni kiriting";
cin>>X>>h;
A[i]=cos(X);
S=0;
for(i=2;i<21;i++)
A[i]=cos(X+(i-1)*h);
cout<<"A[i]="<<A[i];
if(fabs(A[i])>0.5)
S=S+A[i];
cout<<"S="<<S;
return 0; }
```



362. $x=\{x_1,...,x_{15}\}$ vektorning komponentalar qiymati quyidagi formula

asosida hisoblansin:

$$x_i = 13\cos(0,1i + \sin 2i).$$

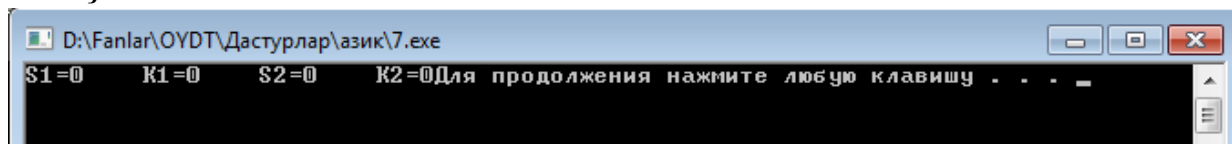
Hosil bo'lgan vektor chop etilsin. $[-0,5; 0,5]$ kesmaga tegishli x vektorning komponentalar yig'indisi, bunday komponentlar soni, shuningdek $[0,5; 1]$ kesmaga tegishli komponentlar yig'indisi va bunday komponentlar soni hisoblansin.

```
#include <iostream>
#include <math.h>
using namespace std;
int main(int argc, char *argv[]){
```

```

int i;
float X[15],S1,S2;
int K1,K2;
S1=0;
S2=0;
K1=0;
K2=0;
for(i=1;i<15;i++) {
X[i]=1.3*cos(0.1*i)+sin(2*i);
if(X[i]>=-0.5 && X[i]<=0.5){
S1=S1+X[i];
K1=K1+1; }
if(X[i]>=0.5 && X[i]<=1){
S2=S2+X[i];
K2=K2+1; } }
cout<<"S1="<<S1<<"\t"<<"K1="<<K1<<"\t"<<"S2="<<S2;
cout<<"\t"<<"K2="<<K2;
return 0;
}

```



Mustaqil ishlash uchun vazifa

363. $h = \sum_{i=1}^n (a_i - b_i)$ yig'indini hisoblovchi dasturni tuzing, bunda a_i, b_i quyidagi formulalar asosida hisoblanadi ($n=5$):

$$a_i = \begin{cases} i, & \text{agar } i - \text{toq,} \\ i^2, & \text{agar } i - \text{juft;} \end{cases}$$

$$b_i = \begin{cases} i^2, & \text{agar } i - \text{toq,} \\ i, & \text{agar } i - \text{juft;} \end{cases}$$

Dasturda hisoblab topilgan a va b vektorlar chop etilishi nazorat qilinsin.

364. $m_i, (i=1, \dots, n)$ massali $M_i(x_i, y_i, z_i)$ moddiy nuqtalar sistemasining $M_c(x_c, y_c, z_c)$ og'irlik markazi koordinatalari quyidagi formulalar asosida hisoblaydigan dastur tuzing ($n=5$):

$$x_c = \frac{\sum_{i=1}^n m_i x_i}{\sum_{i=1}^n m_i}, \quad y_c = \frac{\sum_{i=1}^n m_i y_i}{\sum_{i=1}^n m_i}, \quad z_c = \frac{\sum_{i=1}^n m_i z_i}{\sum_{i=1}^n m_i},$$

365. $x = \{x_1, \dots, x_{10}\}$ vektorni hisoblaydigan dastur tuzing. X vektorning har bir komponentasi quyidagi formula asosida topiladi;

$$x_j = \begin{cases} \arctg \frac{\sqrt{j+2}}{n+3}, & tgj \leq 3, \\ e^{j+\cos n}, & tgj > 3, \end{cases} \quad j=1, \dots, 10, n=10.$$

366. 0.1 qadam bilan, 0 dan 1gacha o'zgaruvchi x da $y = 2\sin x + \frac{1}{\cos x}$ qiymatlarini hisoblovchi va chop etuvchi dasturni tuzing. x va y qiymatlarini hisoblashni vektor ko'rinishida tasvirlangan va ekranda chop eting.

367. 0.2 qadam bilan, 0.1 dan 1.9 gacha o'zgaruvchi x da quyidagi funksiyani hisoblovchi chop etuvchi dastur tuzing.

$$y = 2\arccos \frac{x}{2} + x \ln \left| tg \frac{x}{2} \right|.$$

x va y qiymatlarini hisoblashni vector ko'rinishida tasvirlang va ekranda chop eting.

368. Quyidagi tenglamani yechadigan dasturni tuzing.

$$a_i x + b_i = 0, \quad i = 1, \dots, 5; j = 1, \dots, 4.$$

$a(a_1, \dots, a_5)$ va $b(b_1, \dots, b_4)$ massivlar berilgan.

$a_i \neq 0$ bo'lganda yechim chop etilsin, $a_i = 0$ bo'lganda m o'zgaruvchiga 0 qiymat berilsin va u chop etilsin.

369. Berilgan n dan N kattalikni quyidagi formula asosida hisoblovchi dastur tuzing:

$$N = \begin{cases} n!, & n \leq 3, \\ n^8, & n > 10. \end{cases}$$

Keyin esa quyidagi

$$F = lg \sqrt{N}$$

Hisoblansin va N hamda F qiymatlari chop etilsin.

370. $N = (m+n)!$ kattalikni hisoblovchi dasturni tuzing (butun m va n lar berilgan). Keyin esa

$$F = 1.2 \cos m + 0.8 \sin n) \sqrt{lg N}$$

Hisoblansin. N va F chop etilsin.

371. n -o'lchovli $x = \{x_1, \dots, x_n\}$ vektorning uzunligini quyidagi formula asosida hisoblovchi dasturi tuzing.

$$L = \sqrt{x_1^2 + x_2^2 + \dots + x_n^2}.$$

x vektor berilgan, $n=4$.

372. $a = \{a_1, \dots, a_n\}$ vektoni hisoblovchi dasturni tuzing, uning har bir komponenti berilgan formula asosida aniqlanadi.

$$x_j = \begin{cases} \sin \frac{i^2+1}{n}, \sin \frac{i^2+1}{n} > 0, \\ \cos(i + \frac{1}{n}), \sin \frac{i^2+1}{n} \leq 0, \end{cases} \quad i=1, \dots, n; \quad n=10.$$

373. Berilgan formula asosida aniqlanadigan m matematik kutilmani

$$m = \frac{1}{n} \sum_{i=1}^n x_i,$$

va cheklanishning o'rtacha modulini

$M = \frac{1}{n} \sum_{i=1}^n |x_i - m|$ hisoblovchi dasturni tuzing. $x = \{x_1, \dots, x_n\}$ tanlanish berilgan, $n=11$.

374. Quyidagilarni hisoblovchi dasturni tuzing:

$$S = \sum_{x_i < 0} |x_i|, \quad F = \sum_{x_i \geq 0} |x_i|,$$

bunda $x = \{x_1, \dots, x_n\}$ - berilgan vektor.

375. $z = \{z_1, \dots, z_n\}$ vektor komponentiari modullarining yig'indisini hisoblovchi dasturni tuzing. Z vektor komponentiari qiymati quyidagi formula asosida hisoblanadi:

$$z_i = \begin{cases} \sin i, & \cos i < 0,5, \\ \operatorname{ctg} i^2, & \cos i \geq 0,5, \end{cases} \quad i = 1, \dots, 9$$

Olingan Z vektor chop etilsin.

376. $c = \{c_1, \dots, c_6\}$ vektorni hisoblovchi dasturni tuzing, uning har bir komponenti

$$c_k = a_k + kb_k$$

formula asosida topiladi.

a vektor berilgan. B vektor komponentiari qiymatini quyidagi formula asosida hisoblang:

$$b_k = \sqrt{\left| \arctg\left(\frac{k}{2} - 1.5\right) \right|}, \quad k=1, \dots, 6$$

B va c vektorlarni chop eting.

377. Quyidagi kattalikni hisoblovchi dasturni tuzing:

$$P = \prod_{k=1}^3 (1 - x^k),$$

Jumladan, x^k kattalikni hisoblash uchun $x^k = x^{k-1} \cdot x$ rekurent munosabat qo'llanilsin.

378. n -o'lchovli fazoda ikkita nuqta $P_1(x_1, x_2, \dots, x_n)$ va $P_2(y_1, y_2, \dots, y_n)$ orasidagi masofani hisoblovchi dastumi tuzing.

$$d = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \dots + (x_n - y_n)^2}$$

masofani hisoblovchi dastumi tuzing.

379. x ning 0,25 qadam bilan —2 dan to 2 gacha o'zgaruvchi quyidagi funksiyani hisoblovchi va chop etuvchi dasturni tuzing:

$$z = \begin{cases} 1, & |x| + |y| \leq 1, \quad x \geq 0, \\ \sin(x + y), & |x| + |y| > 1, \quad x < 0, y < 0, \\ e^{x^2 + |y|} & q \text{ lganhollarda} \end{cases}$$

y ning qiymati ko'rsatilgan.

380. Quyidagi masalani yechadigan dastur tuzing: $y = ax + b$ to'g'ri chiziq va $x^2 + y^2 = r^2$ aylana berilgan.

S o'zgaruvchiga 0, 1 yoki 2 qiymatlari mos ravishda to'g'ri chiziq va aylana 2 ta nuqtada kesishmaganda, uringanda yoki kesishganda berilsin.

381. Quyidagi approksimlashtirilgan ko'phadning koeffitsiyentlarini hisoblash uchun dastur tuzing:

$$P(x) = a_1 + a_2 x.$$

Ko'phadning a_1 va a_2 koeffitsiyentlari berilgan tenglamalar sistemasi orqali topiladi:

$$\begin{aligned} S_1 a_1 + S_2 a_2 &= t_1 \\ S_2 a_1 + S_3 a_2 &= t_2, \end{aligned}$$

bu yerda

$$\begin{aligned} S_k &= \sum_{i=1}^n x_i^{k-1}, \quad k = 1, 2, 3, \\ t_j &= \sum_{i=1}^n x_i^{j-1}, \quad j = 1, 2. \end{aligned}$$

$x = \{x_1, \dots, x_n\}$ va $y = \{y_1, \dots, y_n\}$ qiymatlar berilgan ($n \leq 20$).

383. Quyidagi kattalikni hisoblovchi dasturni tuzing.

$$N = \max_i \sum_{j=1}^n |x_{ij}|, j = 1, 2, \dots, n$$

bu yerda $A = \{x_{ij}\}$, $i, j=1, 2, \dots, n$ matrisa berilgan.

384. $S = \{s_1, \dots, s_m\}$ vektorning eng kichik komponentini topish uchun dastur tuzing, bu yerda har bir S_j komponenta quyidagi formula asosida topiladi:

$$S_k = \sum_{i=1}^n a_{ij}, j = 1, 2, \dots, m.$$

$A = \{a_{ij}\}$, $i=1, 2, \dots, n$, $j=2, \dots$, matritsa elementlari berilgan.

385. Quyidagi kattalikni hisoblovchi dasturni tuzing.

$$S = \sum_{i=1}^n a_k z(t_k),$$

Bu yerda t_k argumentli z funksiya chiziqli interpolyasiya formulasi orqali hisoblab topiladi:

$$z(t_k) = y_i + \frac{y_{i+1} - y_i}{x_{i+1} - x_i} (t_k - x_i)$$

Jumladan $x_i \leq t_k \leq x_{i+1}$, $i = 1, 2, \dots, n$, $k = 1, 2, \dots, m$.

386. t argument qiymati uchun z funksiya qiymatini quyidagi formula asosida hisoblang:

$$z = \sum_{i=1}^n a_k u_k(t),$$

Bu yerda $u_{k+1}(t) = 2tu_k(t) - u_{k-1}(t)$, $u_1(t) = 1$, $u_2(t) = t$, $k = 2, 3, \dots, n-1$. $t, a_1, \dots, a_n$ qiymatlar berilgan.

387. a_k ($k = 1, 2, \dots, n$) koeffitsiyentlami quyidagi formula asosida hisoblovchi dasturni tuzing:

$$a_k = \frac{4}{n\pi} \sum_{i=1}^n \frac{y_i u_k(x_i)}{\sqrt{1 - x_i^2}}, \quad k = 1, 2, \dots, n,$$

Bu yerda. $u_{k+1}(x_i) = 2tu_k(x_i) - u_{k-1}(x_i)$, $u_1(x_i) = 1$, $u_2(x_i) = x_i$, $k = 2, 3, \dots, n-1$.

388. Quyidagi masalani yechadigan dastur tuzing. Berilgan $y = \{y_1, y_2, \dots, y_n\}$ vektorni $A = \{a_{ij}\}$ matrisaning birinchi ustuni deb

hisoblab, ya'ni $a_n = y_i$ ($i=1, 2, \dots, n$), matrisaning qolgan elementlarini ko'rsatilgan formula orqali hisoblang:

$$a_{ij} = \begin{cases} a_{i+1,j+1}, & i+j-1 \leq n, \\ 0, & i+j-1 > n \end{cases}$$

Keltirilgan formulaga asosan $a_{n1}, \dots, a_{1n}$ elementlar joylashgan diagonaldan quyida turgan matrisa elementlari nolga aylanadi.

389. M matematik kutilmani va D dispersiyani quyidagi formulalar asosida hisoblang:

$M = \frac{1}{n} \sum_{i=1}^n a_i$; $D = \frac{1}{n} \sum_{i=1}^n a_i^2 - M^2$; tanlanishni tashkil etuvchi n ta $a_1, a_2, \dots, a_n$ sonlarquyidagi formula asosida hisoblanadi

$$a_i = \begin{cases} \sin i, & i > 17 \\ \operatorname{ctg} i^2, & i \leq 17, \end{cases} \quad i=1, 2, \dots, n.$$

390. $x_1, x_2, \dots, x_n$ nuqtalarda $y(x)$ funksiya hosilasini sonli differensiyalash formulalari orqali hisoblovchi dasturni tuzing:

$$y'(x)|_{x_1} = y' = \frac{-3y_1 + 4y_2 - y_3}{2(x_2 - x_1)},$$

$$y'(x)|_{x_i} = y' = \frac{y_{i+1} - y_{i-1}}{2(x_{i+1} - x_i)}, \quad i = 2, 3, \dots, n-1$$

$$y'(x)|_{x_n} = y' = \frac{y_{n-2} - 4y_{n-1} + 3y_n}{2(x_n - x_{n-1})},$$

$y(x)$ funksiya jadval sifatida berilgan, ya'ni $x_1, x_2, \dots, x_n$ argument qiymatiga berilgan $y_1, y_2, \dots, y_n$ funksiya qiymatlari mos keladi.

391. Qator yig'indisini hisoblovchi dasturni tuzing.

$$S = 1 - \frac{1}{2} \sum_{n=0}^{\infty} 2^n (a_n^2 - b_n^2),$$

bu yerda $a_n = \frac{1}{2}(a_{n-1} + b_{n-1})$, $b_n = \sqrt{a_{n-1}b_{n-1}}$. a_0 va b_0 qiymatlari berilgan. Hisoblashni $a_n^2 - b_n^2 < q$, $q=10^{-6}$ shart bajarilganida to'xtatish kerak.

392. Tartiblangan $\{x_1, x_2, \dots, x_n\}$ va $\{y_1, y_2, \dots, y_n\}$ sonlar ketma-ketligi orasida bog'liqlik me'yorini ifodalovchi korrelyasiya koeffitsiyentini berilgan formula asosida hisoblovchi dasturni tuzing:

$$r = \frac{S_{11}S_{22} - S_{21}S_{12}}{\sqrt{(S_{11}S_{31} - S_{21}^2)(S_{11}S_{13} - S_{12}^2)}}$$

Bu yerda $S_{kj} = \sum_{i=1}^n x_i^k y_i^j$ $k=1,2,3, j=1,2,3$.

393. $A = \{a_{ij}\}$ matrisa elementlarini hisoblovchi dasturni tuzing, uning har bir element!

$$a_{ij} = \begin{cases} b_{ij}, & |b_{ij}| > |c_{ij}| \\ c_{ij}, & |b_{ij}| \leq |c_{ij}| \end{cases} \quad (i = 1, 2, \dots, m)$$

formula asosida hisoblanadi. $B=\{b_{ij}\}$ $C=\{c_{ij}\}$ matrisalar berilgan.

394. Berilgan yig'indini hisoblovchi dasturni tuzing:

$$S = \sum_{k=2}^{n+1} a_k,$$

Bu yerda $a_k = \frac{1}{(2k)!} \prod_{i=1}^{k-1} x_i^2, \quad k = 2, 3, \dots, n+1. \quad x_1, x_2, \dots, x_n$

kattalik qiymati berilgan.

395. Nyuton binomini hisoblovchi dasturni tuzing:

$$(1+x)^m = 1 + \sum_{i=1}^m c_m^i x^i$$

c_m^{i+1} ni hisoblash uchun quyidagi rekurrent munosabatdan foydalanilsin:

$$c_{i+1} = c_i \frac{m-i}{i+1}, \quad c_i = m, \quad i=1, 2, \dots, m-1 \quad (c_i \text{ orqali } c'_m \text{ ifodalangan}).$$

396. $S=\{s_{ij}\}$ ($i, j=1,2,3$) matrisani hisoblovchi dasturni tuzing. Har bir s_v element quyidagi formula asosida hisoblanadi:

$$S_{ij} = \sum_{k=1}^n x_k^{i-1} y_k^{j-1}, \quad i, j = 1, 2, 3, \text{bu yerda}$$

$$x_k = \begin{cases} a_k, & a_k \geq 0 \\ -1, & a_k < 0 \end{cases}, \quad y_k = \begin{cases} a_k, & a_k < 0 \\ 1, & a_k \geq 0 \end{cases}, \quad (k = 1, 2, \dots, n).$$

$a_1, a_2, \dots, a_n$ qiymatlar berilgan.

397. $z = \{z_1, z_2, \dots, z_n\}$ vektorni hisoblovchi dasturni tuzing, uning har bir komponenti $z_k = x_k + m y_k$ formula orqali aniqlanadi, bu yerda $x_k, y_k \quad x = \{x_1, x_2, \dots, x_n\}$ va $y = \{y_1, y_2, \dots, y_n\}$ vektorlarning komponentlari,

$$m = \begin{cases} k, & |sink| \leq 0.2, \\ \sqrt{k}, & 0.2 \leq |sink| \leq 0.9 \\ \sqrt{\sqrt{k}}, & |sink| \geq 0.9 \end{cases}$$

398. Uchlari $P(x_1, y_1), P_2(x_2, y_2), \dots, P_n(x_n, y_n)$ bo'lgan ko'pburchakning yuzini quyidagi formula asosida hisoblovchi dasturni tuzing:

$$S = \frac{1}{2} [(x_1 - x_2)(y_1 + y_2) + (x_2 - x_3)(y_2 + y_3) + \dots + (x_n - x_{n-1})(y_n + y_{n-1})]$$

bu yerda $x_{n+1} = x_1, y_{n+1} = y_1$.

399. $a = \{a_1, a_2, \dots, a_n\}$ va $b = \{b_1, b_2, \dots, b_n\}$ vektorlar orasidagi φ burchakni hisoblovchi dasturni tuzing, burchakni quyidagi munosabatda aniqlasa bo'ladi:

$$\cos \varphi = \frac{\sum_{i=1}^n a_i b_i}{\sqrt{\sum_{i=1}^n a_i^2 \sum_{i=1}^n b_i^2}}$$

400. Quyidagi ifodani hisoblovchi va chop etuvchi dasturni tuzing;

$$L = \alpha \sqrt{\prod_{i=1}^n \sum_{j=1}^n a_{ij}^2},$$

bu yerda α - x, y, z sonlar orasida eng kattasi, $A\{a_{ij}\}$ - berilgan matritsa.

401. Bichizikli shaklni hisoblovchi dasturni tuzing:

$$B = m \sum_{i=1}^n x_i \sum_{k=1}^n a_{ik} y_k,$$

bu yerda $m \cdot x = \{x_1, x_2, \dots, x_n\}$ vektorning eng katta elementi. $A = \{a_{ij}\}$ $i, j = 1, 2, \dots, n$ matritsa, $x = \{x_1, x_2, \dots, x_n\}$ vektor va $y = \{y_1, y_2, \dots, y_n\}$ vektor berilgan.

402. $a = \{a_1, \dots, a_{30}\}$ komponentlar qiymati quyidagi fonnula asosida hisoblansin:

$$a_i = \sin \frac{i^2 + 1}{30} \quad (i = 1, \dots, 30).$$

Hosil bo'lgan vektor chop etilsin. So'ngra, quyidagi yig'indilar hisoblansin;

$$S_1=a_1+a_6+a_{11}+\dots+a_{26},$$

$$S_2=a_2+a_7+a_{12}+\dots+a_{27},$$

.....

$$S_5=a_5+a_{10}+a_{15}+\dots+a_{30}.$$

Hosil bo'lgan $S = \{S_1, \dots, S_5\}$ vektor chop etilsin.

403. 0,05 qadam bilan 1 dan 2 gacha o'zgaruvchi x qiymatlar uchun

$$xy=e^{-y}$$

tenglama bilan aniqlanadigan noaniq $y(x)$ funksiya qiymatlari jadvalini hisoblash uchun dasturni tuzing.

Har bir x da tenglamani, 10^6 dan oshmaydigan absolyut xatolikli Nyuton uslubi bilan yeching. $X=1$ da boshlang'ich yaqinlashish sifatida 0 (Nyuton uslubining yaqinlashishini tekshiring) qabul qilinsin, qolgan x larda esa boshlang'ich yaqinlashish sifatida oldingi x uchun hisoblangan y qiymat ishlatilsin. Hosil qilingan x va y qiymatlarini vektor sifatida tasvirlang va chop eting.

404. $a = \{a_1, \dots, a_{10}\}$ vektor komponentlari absolyut kattalik bo'yicha o'sib borish tartibida joylashgan. K -butun va r -haqiqiy berilgan. Quyidagi masalani yechadigan dasturi tuzing: a dan k -chi komponentni olib tashlab, b vektordagi komponentlar ham absolyut kattalik bo'yicha o'sib borish tartibida joylashishi uchun r komponentani kerakli joyga qo'yib $b = \{b_1, \dots, b_{10}\}$ vektor hosil qilinsin va chop etilsin.

405. Tekislikda r ichki radiusli va R tashqi radiusli aylanasimon halqaning markazidan $S_1, S_2, \dots, S_{15}$ masofada nuqtalar joylashgan. Halqaning ichida joylashgan nuqtalar sonini aniqlang.

406. $p = \{p_1, \dots, p_n\}$ vektordan $r = \{r_1, \dots, r_m\}$ ($m \leq n$) vektorni quyidagi qoida asosida ajratuvchi dasturni tuzing: Agar

$$x^2 - 2p_i x + q = 0$$

kvadrat tenglama haqiqiy va har xil ildizlarga ega bo'lsa, p vektor komponenti r vektoring komponenti bo'ladi.

407. Berilgan m tartibli kvadrat matrisaning normasi quyidagi formula asosida hisoblansin:

$$N = \max_i \sum_{k=1}^m |a_{ik}|.$$

408. Berilgan m tartibli matrisaning normasi quyidagi formula asosida hisoblansin:

$$N = \max \sum_{k=1}^m |a_{ik}|.$$

409. $x = \{x_1, \dots, x_6\}$ haqiqiy vektor hosil qilish dasturini quyidagi qoida asosida tuzing: agar $x_1 \leq x_2 \leq \dots \leq x_6$ bo'lsa, unda barcha komponentlarga ularning eng kattasining qiymati berilsin. Agar $x_1 > x_2 > \dots > x_6$ bo'lsa, unda vektor o'zgartirilmasin. Aks holda barcha komponentlarni ularning kvadratlari bilan almashtiring. X vektor komponentlarini oldindan quyidagi formula asosida hisoblang:

$$x_i = \arctg(\sqrt{i} + \frac{2}{9}), i = 1, \dots, 6.$$

410. $x = \{x_1, \dots, x_n\}$ vektor komponentlarini quyidagi formula asosida hisoblang:

$$x_i = 4 \arctg(1 + 0,1i) + e^{-0,5i} - 2\pi, \quad i = 1, \dots, n$$

Hisoblangan vektor chop etilsin. Hosil bo'lgan vektorni quyidagi qoida asosida o'zgartiring: barcha manfiy elementlarni 0.5 ga kattalashtiring, barcha musbatlarni esa -0.1 ga almashtiring. Hosil bo'lgan vektorni ham chop eting.

411. $a = \{a_1, \dots, a_n\}$ mantiqiy va $x = \{x_1, \dots, x_n\}$ haqiqiy vektor berilgan. X vektorni quyidagi qoida asosida o'zgartiring: agar a_i TRUE qiymatga ega bo'lsa, unda x_i ni 10 ga kattalashtiring, aks holda x_i ni nol bilan almashtiring. Hosil bo'lgan x vektor chop etilsin.

412. $x = \{x_1, \dots, x_n\}$ vektor komponentlari quyidagi formula asosida hisoblansin:

$$x_i = \cos\left(i + \frac{1}{n}\right), \quad i = 1, \dots, n$$

Hosil bo'lgan vektor chop etilsin. X vektorning manfiy komponentlari va $[0,5;1]$ kesmaga tegishli bo'lgan komponentlar hisoblansin. Berilgan kesmaga tegishli bo'lmagan manfiy komponentlarni 1 bilan almashtiring. Qolgan komponentlarni o'zgartirmang. Hosil bo'lgan vektor chop etilsin.

413. $x = \{x_1, \dots, x_n\}$ vektor komponentlari qiymati quyidagi formula asosida hisoblansin:

$$x_i = \text{arcctg}(\sqrt{i} + \frac{1}{n}) + e^{-0,5i} \quad i=1, \dots, n.$$

Hosil bo'lgan vektor chop etilsin. Agar x vektor komponentlari o'sib boruvchi ($x_1 < x_2 < \dots < x_n$) ketma ketlikni hosil qilsa, unda mantiqiy o'zgaruvchiga TRUE, aks holda FALSE qiymati berilsin.

414. Haqiqiy matrisa berilgan.

$$\{a_{ij}\}, i, j=1, 2, \dots, n. \quad n \leq 15.$$

Asosiy diagonalda manfiy elementi qatorda joylashgan matrisa elementlari yig'indisi hisoblansin. (Asosiy diagonal $a_{ii} (i = 1, \dots, n)$ element tashkil topgan).

415. $x = \{x_1, \dots, x_n\}$ vektor berilgan. Bu vektor komponentlarining siklik surilishi amalga oshirilsin.

a) Chapga bitta pozitsiya, yani quyidagi vektor hosil qilinsin:

$$x = \{x_2, x_3, \dots, x_k, x_1\};$$

b) O'ngga ikkita pozitsiya, yani quyidagi vektor hosil qilinsin:

$$x = \{x_{k-1}, x_k, x_1, \dots, x_{k-2}\}.$$

Boshlang'ich va ikkala hosil bo'lgan vektor chop etilsin.

416. $x = \{x_1, \dots, x_n\}$ vektor qiymatlari quyidagi formula asosida hisoblansin:

$$x_i = 1,5 \lg 0,5i, \quad i=1, \dots, 10.$$

Hosil bo'lgan x vektor chop etilsin.

X va berilgan y vektorlar quyidagi qoida assosida o'zgartirilsin:

x_i, y_i kattasi yangi x_i qiymat sifatida qabul qilinsin,

$kichigi$ esa yangi $y_i, i=1, \dots, 10$ qiymat sifatida qabul qilinsin.

Hosil bo'lgan x, y vektorlar chop etilsin.

417. $x = \{x_1, \dots, x_n\}$ haqiqiy vektorlar berilgan. $y = \{y_1, \dots, y_n\}$ vektor komponentlari quyidagi formula asosida hisoblansin:

$$y_i = 0.26 \ln(1 + \sqrt{i})$$

Agar

$$S = (x_i y_n)^2 + ((x_2 y_{n-1})^2 + \dots + (x_n y_1)^2$$

$[0;1]$ kesma tegishli bo'lsa mantiqiy o'zgaruvchiga TRUE qiymati, aks holda FALSE qiymati berilsin:

X, y vektorlar, hamda S ning qiymati chop etilsin.

418. $a = \{a_1, \dots, a_{10}\}$ va $b = \{b_1, \dots, b_{10}\}$ vektorlar komponentlarining qiymatini quyidagi formula asosida hisoblang:

$$a_i = \arctg \frac{i+2}{10}, \quad b_i = e^{i+\cos \sqrt{10}}$$

Hosil bo'lgan vektorlar chop etilsin. a va b vektorlardan komponentlari 1 dan 20 gacha qayta nomerlashtirilgan $c = \{a_1, b_1, a_2, b_2, \dots, a_{10}, b_{10}\}$ vektor olinsin. Hosil bo'lgan c vektor chop etilsin.

419. $A = \{a_{ij}\}$, $i, j = 1, \dots, n$ matrisa berilgan. t o'zgaruvchi x va y vektorlarning skalyar ko'paytmasidan hosil qilingan qiymati berilsin. x va y vektorlarning komponentlarni quyidagi usul bilan aniqlanadi:

$$x_i = \max\{a_{ij}\}, \quad y_i = \min\{a_{ki}\}, \quad i = 1, \dots, n$$

420. $a = \{a_1, \dots, a_{10}\}$ haqiqiy sonli vektor berilgan. $b = \{b_1, \dots, b_{10}\}$ vektorni shunday hosil qilinganki, uning birinchi komponentlari sifatida barcha manfiy a vektorlar (ularning joylashish tartibini saqlagan holda), qolgan komponentlar sifatida esa a vektorning manfiy komponentlari olinsin.

421. $a = \{a_1, \dots, a_n\}$ butun sonli vektor berilgan. Agar ushbu vektorning komponentlari orasida b butun songa teng komponenta qiymati bo'lsa, unda l o'zgaruvchiga ushbu komponentdan oldingi barcha komponentlar yig'indisi berilsin. Aks holda l=0.

422. $x = \{x_1, \dots, x_n\}$ vektorlar komponentlar qiymati quyidagi formula asosida aniqlansin:

$$x_i = 1, 2 \sin 0, 1i + 0, 4 \cos(i-1), \quad i = 1, 2, \dots, n.$$

Hosil bo'lgan vektor chop etilsin.

X vektorning eng kattasi topilsin, uning k nomeri chop etilsin. Eng katta x_k komponentdan oldin keluvchi x vektorning juft nomerli barcha komponentlarni x_k ga ko'paytiring. Qolgan komponentlar o'zgartirilmasin. Hosil bo'lgan vektorni chop eting.

423. $x = \{x_1, \dots, x_n\}$ vektor berilgan. Agar ushbu vektorning aqalli bitta komponenti t=-2 dan kichgina bo'lsa, unda barcha manfiy komponentlarni qolgan komponentlarni o'zgarishsiz

qoldirgan holda, ularning kvadratlari bilan almashtiring; aks holda x vektorning barcha komponentlarini 0,1 ga ko'paytiring. Hosil bo'lgan vektor chop etilsin.

424. $A=\{a_{ij}\}$, $i,j=1,\dots,n$ haqiqiy matrisa berilgan. $b = \{b_1, \dots, b_n\}$ haqiqiy vektorni quyidagi qoida asosida yarating: agar k , $k=1, 2, \dots, n$ qatorda musbat elementlar soni manfiy elementlar soniga qaraganda ko'proq bo'lsa, unda b_k ga TRUE qiymat berilsin; aks holda b_k ga FALSE qiymat berilsin.

425. $A=\{a_{ij}\}$, $i,j=1,\dots,n$ haqiqiy matrisa berilgan. $b = \{b_1, \dots, b_n\}$ haqiqiy vektorni quyidagi qoida asosida yarating: agar $a_{ii} < 0$, $i=1,\dots,n$ bo'lshi, unda b_i sifatida i -chi qatordagi a_{ii} dan oldin kelgan elementlar yigindisi qabul qilinsin; agar $a_{ii} \geq 0$ bo'lsa, unda b_i sifatida i -chi qatordagi a_{ii} (a_{ii} ham olinadi) dan oldin kelgan elementlar yig'indisi olinsin;

426. $A=\{a_{ij}\}$, $i,j=1,\dots,n$ haqiqiy matrisa berilgan. Birlamchi hamma elementlari absolyut kattalik bo'yicha eng katta bo'lgan elementga bo'lish orqali yangi matrisa qabul qilinsin.

427. $A=\{a_{ij}\}$, $i,j=1,\dots,n$ haqiqiy matrisa berilgan. $l = \{l_1, \dots, l_n\}$ mantiqiy vektorni quyidagi qoida asosida tuzing: agar A matrisaning i -chi qatori kamayuvchi ketma-ketlikni hosil qilsa, unda l_i ga TRUE qiymat berilsin. Aks holda l_i ga FALSE qiymat berilsin.

428. $a = \{a_1, \dots, a_n\}$ mantiqiy va $x = \{x_1, \dots, x_n\}$ haqiqiy vektorlar berilgan. x vektorni quyidagi qoida asosida hosil qiling: agar a_1 TRUE qiymatga ega bo'lsa, unda x_i ni 10 ga ko'paytiring, aks holda x_i ning belgisi teskarisiga almashtirilsin.

429. $x = \{x_1, \dots, x_n\}$ vektorning komponentlar qiymatini quyidagi formula asosida hisoblang:

$$x_i = \begin{cases} \arctg \frac{\sqrt{i} + 2}{n + 3}, & n \leq 13, \\ \cos i e^{i + \cos n}, & n > 13, \quad i = 1, \dots, n \end{cases}$$

Hosil qilingan vektor chop etilsin. x vektorning musbat va manfiy komponentlari hisoblansin. Hosil qilingan natija chop etilsin.

430. Berilgan $a = \{a_1, \dots, a_n\}$ vektorning komponentlari kamayish tartibida joylashgan. Quyidagi masalani yechadigan

dasturni tuzing: a vektor va r o'zgaruvchidan komponentlari kamayib borish tartibda joylashgan b vektor hosil qilinsin, ya'ni r o'zgaruvchini a vektor komponentlari orasida shunday joylashtiringki $a_i < r < a_{i+1}$ shart bajarilsin. B vektor komponentlari quyidagi usulda olinadi:

$$b_j = a_j \quad (1 \leq j \leq i), \quad b_{i+1} = r,$$

$$b_j = a_{j-1} \quad (i+2 \leq j \leq 10).$$

B vektorning juft komponentlarini kvadrat logarifmlari yig'indisi hisoblansin:

$$S = \lg b_2^2 + \lg b_4^2 + \dots + \lg b_{10}^2.$$

B vektor va S ning qiymati chop etilsin.

FUNKSIYALAR

Qismaniy dastur deb, nom bilan ta'minlangan va alohida usul bilan tashkillashtirilgan mustaqil dastur bo'lagiga (fragmentiga) aytiladi.

Funksiya – aniqlangan harakatni bajarish uchun ism bo'yicha chaqirilishi mumkin bo'lgan dasturning nomlangan mustaqil bo'lagidir. Dastur ichida funksiyaning nomini ko'rsatish bilan uni ishga tushirish bajariladi va bu jarayon unga murojaat deb yuritiladi.

Funksiya tavsifi sarlavha va tanadan iborat. Funksiya sarlavhasi quyidagi ko'rinishda tavsiflanadi:

<tur> <ism>([<ro'yxat>]);

Bu yerda <tur> – funksiya qaytaradigan qiymat turi, agar funksiya xech qanday qiymat qaytarmasa yoki ko'p qiymat qaytarsa **void** kalit so'zi yoziladi; <ism> - funksiya nomi, <ro'yxat> - formal parametrlar ro'yxati (sarlavhada aniqlangan).

Formal parametrlar ro'yhati bo'lmasligi ham mumkin. Agar bor bo'lsa, bir-biridan vergul bilan ajratilgan formal parametrlar turi va ularning ismidan tashkil topadi.

Funksiyaga murojaat qilish dasturning ixtiyoriy joyidan turib amalga oshiriladi va quyidagi ko'rinishga ega bo'ladi:

<ism>([<aniq parametrlar ro'yhati>]);

Funksiya tanasi {} operatorlar qavslariga olinib tavsiflar va operatorlardan tashkil topishi mumkin.

Funksiya qaytadigan qiymat return operatorida ko'rsatiladi.

Funksiya prototipi faqat funksiya sarlavhasidan tashkil topadi.

Bu bo'limdagi masalalarni yechishdan oldin quyidagilarni yodga olish kerak:

1. funksiya ma'lumotlarni o'zlash uchun faqat parametrlardan foydalanish kerak. Global o'zgaruvchilar, ya'ni funksiya tashqarida tavsiflangan o'zgaruvchilardan foydalanish tavsiya etilmaydi:

2. funksiya murojaat ko'rsatmasidagi har bir faktik parametrlar turi (o'zgaruvchilar yoki o'zgaruvchilar), funksiya tavsifida ko'rsatilgan formal parametrlar turi bilan mos kelishi kerak;

3. agar funksiya parametri natijani qaytarish uchun ishlatilayotgan bo'lsa, u holda funksiya tavsifida bu parametr havola bo'lishi kerak, funksiyaning chaqirishdagi ko'rsatmada esa faktik parametr sifatida o'zgaruvchi adresini ko'rsatish kerak.

Misollar

431. Silindr hajmini hisoblaydigan funksiya tuzing. Funksiya parametrlari sifatida silindrning radiusi va balandligi bo'lishi kerak.

```
#include <iostream>
#include <conio.h>
#include <math.h>
// Silindr hajmi
float vcil(float h, float r) {
    return(M_PI*r*r*h);
}
int main() {
    float r,h; // silindr asosi radiusi va balandligi
    float v; // Silindr hajmi
    puts("Silindr hajmini hisoblash");
    printf("Asos radiusi va balandligini kiriting->");
    scanf("%f%f", &h, &r);
    v = vcil(h, r);
    printf("Silindr hajmi %3.2f\n", v);
    printf("Ishni tugatish uchun <Enter> ni bosing");
    return 0;
}
```

432. Argument sifatida olingan ikkita butun son dan eng kattasini qaytaradigan funksiya tuzing.

```
//max funksiyasi ikkita son dan kattasini qaytaradi
int max(int a, int b) {
    if (a > b)
        return(a);
    else
        return (b);
}
```

433. Berilgan ikkita sonni taqqoslaydigan va natijani munosabat belgisi ko‘rinishida qaytaradigan funksiyani e‘lon qiling.

```
//compare funksiyasi sonlarni taqqoslash natijasini
//munosabat belgisi ko‘rinishida qaytaradi
#include <iostream>
#include <math.h>
#include <stdio.h>
#include <conio.h>
char compare(int a, int b) {
    char res;
    if (a > b) res = '>';
    else if (a < b) res = '<';
    else res = '=';
    return(res);
}
int main() {
    int x1,x2; // taqqoslanadigan sonlar
    char res; // taqqoslash natijasi
    puts("Ikkita butun sonni kirit va <Enter> ni bosing");
    printf("->");
    scanf("%i%i", &x1, &x2);
    res = compare(x1,x2); //dasturchi funksiyasiga //murojaat
    printf("%i %c %i\n", x1, res, x2);
    puts("\nIshni tugatish uchun <Enter> ni bosing");
    return 0;
}
```

434. Ekranga ramkani chiqaruvchi frame funksiyani tuzing. Funksiya parametri sifatida ramkaning chap yuqori burchaning koordinatasi va o‘lchami o‘zatiladi.

```
//frame funksiyasi ramkani hosil qiladi
#include <iostream.h>
#include <math.h>
#include <stdio.h>
#include <conio.h>
//ramkani –hosil qilish
void frame(int l, int t, int w, int h) {
```

```

//l,t–ramkaning chap yuqori burchagi
//koordinatalari,
// w, h – uzunligi va balandligi
int x,u; // chiqarilayotgan belgi koordinatasi
int i;
// ramkani hosil qilayotgan belgilar
char s1 == 218, // chap yuqori burchak
s2 = 196,      // gorizontal chiziq
sZ = 191,      // o‘ng yuqori burchak
s4 = 179,      // vertikal chiziq
s5 = 192,      // chap quyi burchak
c6 = 217;      // o‘ng quyi burchak gotoxy(l,t);
putch(c1);
for (i = 0; i < w-2; i++) // ramkaning yuqori // chegara belgilari
putch(s2);
putch(c3);
u = t+1;
x = l+w-1;
for(i=0; i<h-1;i++)//chap va o‘ng chegara //belgilari
{
gotoxy(l,y);
putch(s4);
gotoxy(x,u);
putch(s4);
u++;
}
gotoxy(l,u);
putch(s5);
for (i = 0; i < w-2; i++) // quyi chegara // belgilari
putch(c2);
putch(c6);
}
void main () {
clrscr();
frame(5,5,30,10);
puts("\nIshni tugatish uchun <Enter> ni bosing");

```

```
getch();  
}
```

435. Klaviaturadan musbat butun sonni kiritishni ta'minlaydigan funktsiyani tuzing. Belgiga mos tugmani bosgandan keyin, agar bu belgi raqam hisoblangan holdagina ekranda hosil bo'lishi kerak. Funktsiya kiritilayotgan sonni <Backspace> tugmasi yordamida tahrirlashga ruxsat berishi kerak. <Enter> tugmasi bosilgandan keyin funktsiya ishni tugatishi va kiritilgan sonni qaytarishi kerak.

```
// getint funksiyasi  
#include <iostream>  
#include <math.h>  
#include <stdio.h>  
#include <conio.h>  
#include <stdlib.h>  
  
// getint funksiyasi bitta yoki ikkita raqamdan  
// iborat musbat butun sonni kiritish uchun  
// mo'ljallangan.  
// Kiritish vaqtida tahrirlash uchun <Backspace>  
// tugmasidan foydalanish mumkin.  
// <Enter> ni bosishda funktsiya kiritilgan sonni  
// qaytaradi.  
#define K_VASK 8 // <Backspace> tugmasi kodi  
#define K_ENTER 13 // <Enter> tugmasi kodi  
#define NV 4 // ruxsat etilgan raqamlar soni  
int getint() {  
    char ch;    // joriy belgi  
    char buf[NV]; // raqamlarni kiritish  
    int n = 0;  // kiritilgan raqamlar soni  
    buf[0] = '\0';  
    while ((ch = getch()) != K_ENTER)
```

```

if (ch >= '0' && ch <= '9' && n < NV) {
    putchar(ch);
    buf[n++] = ch;
}
else if (ch == K_VASK && n) {
    printf("\b \b");
    n--;
}
if (n) {
    buf[n] = '\0';
    return(atoi(buf));
}
else return(-1);
}

int main () {
    int a; // kiritilgan son
    puts("\ngetint funksiyasini ishlashini tekshirish\n");
    puts("getint funksiyasi musbat butun sonlarni ");
    puts("kiritish uchun mo'ljallangan.");
    puts("Kiritish vaqtida, tahrirlash uchun, ");
    puts("<Backspace>tugmasidan foydalanish mumkin");
    puts("<Enter> bosilgandan keyin funksiya kiritilgan");
    puts("soni yoki agar son kiritilmagan bo'lsa -1 ni qaytaradi.");
    puts("Sonni kiriting va <Enter> ni bosing");
    printf("->");
    if (a = getint() )
        printf("\nSiz %d sonni kiritdingiz", a);
    else puts("Son kiritilmadi.");
}

```

```
puts("\nIshni tugatish uchun <Enter> ni bosing");
getch();
}
```

Mustaqil ishlash uchun vazifa

436. Ikkita butun sonni taqqoslaydigan va taqqoslash natijasini quyidagi belgilardan biri ko‘rinishida qaytaradigan funksiya tuzing: $>$, $<$ yoki $=$.

437. Ikkita rezistordan tashkil topgan zanjirning qarshiligini hisoblaydigan funksiya tuzing. Qarshiliklar kattaliklari va ulanish turi (ketma-ket yoki parallel) funksiya parametrlari hisoblanadi. Funksiya parametrlarni to‘g‘riligini tekshirishi kerak: agar ulanish turi noto‘g‘ri ko‘rsatilgan bo‘lsa, u holda funksiya -1 ni qaytarishi kerak.

438. a^b ni qiymatini hisoblaydigan funksiya tuzing. a va b sonlar ixtiyoriy musbat kasr sonlar bo‘lishi mumkin.

439. Argument sifatida olinganlardan foizni qaytaradigan Foiz funksiyasini tuzing.

440. “Faktorial” funksiyasini va faktoriallar jadvalini chiqarish uchun bu funksiya dan foydalanadigan dastur tuzing.

441. Omonotning daromatini hisoblaydigan Daromat funksiyasini tuzing. Funksiya uchun boshlang‘ich ma’lumotlar omonot o‘lchami, foiz miqdori (yillik) va omonot muddati (kunlarda) hisoblanadi.

442. Agar funksiya argumenti sifatida olingan belgi krill alifbosidagi unli harf bo‘lsa 1 ni, aks holda 0 ni qaytaradigan unli funksiyasini tuzing.

443. Agar funksiya argumenti sifatida olingan belgi rus alifbosidagi undosh harf bo‘lsa 1 ni, aks holda 0 ni qaytaradigan undosh funksiyasini tuzing.

444. Argument sifatida olingan satrni yuqori registrga o‘tkazilganini qaytaradigan funksiya tuzing.

445. Kvadrat tenglamani yechimini ta’minlaydigan funksiya tuzing. Funksiya parametrlari koeffitsiyentlar va tenglama ildizlari bo‘lishi kerak. Funksiya qaytaradigan qiymat, chaqirgan dasturga

tenglamaning ildizi mavjudligi haqida xabar berishi kerak: 2 – ikkita turli xil ildizlar, 1 – bir xil ildizlar, 0 – tenglama yechimga ega bo‘lmasa. Bundan tashqari funksiya boshlang‘ich ma’lumotlarni to‘g‘riligini ham tekshirishi kerak. Agar boshlang‘ich ma’lumotlar noto‘g‘ri bo‘lsa, u holda funksiya -1 ni qaytarishi kerak.

446. Ekranga yulduzchadan tashkil topgan satrni chiqaruvchi funksiyanı tuzing. Satr uzunligi (yulduzchalar soni) funksiya parametri hisoblanadi.

447. Bir xil belgidan tashkil topgan satrni chiqaruvchi funksiyanı tuzing. Satr uzunligi va belgi prosedura parametri hisoblanadi.

448. Parallelepiped sirtining yuzasini va hajmini hisoblaydigan funksiyanı tuzing.

449. Klaviaturadan kasr sonni kiritishni ta’minlaydigan funksiyanı tuzing. Belgiga mos tugmani bosgandan keyin, agar bu belgi berilgan o‘rindagi ruxsat etilgan hisoblangan holdagina ekranda hosil bo‘lishi kerak. Masalan, funksiya nuqta va birinchi o‘rinda bo‘lmasa minus belgisini bittadan ortiq kirtishga ruxsat bermasligi kerak. Funksiya kiritilayotgan sonni <Backspace> tugmasi yordamida tahrirlashga ruxsat berishi kerak. <Enter>tugmasi bosilgandan keyin funksiya ishni tugatishi va kiritilgan sonni qaytarishi kerak.

450. “21” o‘yinni tashkil etuvchi dastur tuzing. O‘yinchi va kompyuterning kartalarni navbatma-navbat berish bo‘yicha harakatlari funksiyalar ko‘rinishida tashkil etilishi kerak.

SATRLAR VA SATRLAR USTIDA AMALLAR BAJARISH

Standart kutubxonadagi string sinfi. C++ da satrlar bilan ishlashni qulaylashtirish uchun string sinfi kiritilgan. string sinfi satrlarida satr oxirini '\0' belgisi belgilamaydi. string sinfidan foydalanish uchun qaysi sarlavha faylini dasturga qo'shish kerak?

Standart kutubxonadagi string sinfidan foydalanish uchun <string> sarlavha faylini dasturga qo'shish kerak. Lekin ba'zi eski kompilyatorlarda <cstring.h> yoki <bstring.h> sarlavha faylini qo'shish kerak bo'ladi. Oddiy eski usuldagi satrlar bilan ishlash uchun esa, <string.h> sarlavha fayli qo'shiladi.

Eng afzali, o'zingiz ishlatayotgan kompilyator bilan yaxshilab tanishib chiqing. Satrlar bilan ishlovchi asosiy funksiyalar bilan tanishib chiqamiz.

Satr xususiyatlarini aniqlash uchun quyidagi funksiyalardan foydalanish mumkin:

unsigned int size() const; - satr o'lchami.

unsigned int length() const; - satr elementlar soni.

unsigned int max_size() const; - satrning maksimal uzunligi.

unsigned int capacity() const; - satr egallagan xotira hajmi.

bool empty() const; - satrning bo'shligini aniqlash.

Satrning uzunligini aniqlash uchun length() yoki size() funksiyalaridan foydalanish mumkin.

Satr uzunligini o'zgartirish uchun *resize* funksiyasidan foydalaniladi

1) **void** resize (size_t n, **char** c);

2) **void** resize (size_t n);

void clear(); - funksiyasi satrni tozalash (to'liq o'chirish) uchun ishlatiladi.

bool empty() **const;** - funksiyasi satrni bo'shligini tekshirish uchun ishlatiladi.

Agar satr bo'sh bo'lsa, true qiymat qaytaradi.

Satrning biror qismidan nusxa olish

string& assign (**const** string &str);

Satrga str o'zgaruvchisidagi satrning to'liq nusxasini olish.

string& assign (**const** string& str, size_t pos, size_t n);

Satrga str o'zgaruvchisidagi satrning pos o'rindagi belgisidan boshlab n ta belgi nusxasini olish.

string& assign (**const char*** s, size_t n);

string toifasidagi satrga char toifasidagi satrning n ta belgisi nusxasini olish.

string s1, s2, s3;

s1 = "Samarqand";

s2.assign(s1); // s2 = "Samarqand"

s3.assign(s1, 0, 6); // s3 = "Samarq"

append funksiyasining assigndan farqi satrning davomiga satr qismining qo'shishidadir.

string& append (**const** string& str);

string& append (**const** string& str, size_t pos, size_t n);

string& append (**const char*** s, size_t n);

Satrning biror qismini o'chirish:

erase(**unsigned int** pos=0, **unsigned int** n=npos);

erase funksiyasi satrni pos o'zgaruvchisida ko'rsatilgan o'rindan boshlab n ta belgini o'chiradi.

Agar nechta belgi o'chirilishi n ko'rsatilmagan bo'lsa, pos o'zgaruvchisida ko'rsatilgan o'rindan boshlab satr oxirigacha o'chiriladi. Agar pos va n ko'rsatilmagan bo'lsa, satr to'liq o'chiriladi.

Satrni satr orasiga qo'shish. Biror satrga boshqa satrning istalgan qismini qo'shish uchun insert funksiyasidan foydalaniladi.

insert(**unsigned int** pos1, **const** string &str);

Satrga pos1 o'rindan boshlab, str satrini qo'shish.

insert(**unsigned int** pos1, **const** string &str, **unsigned int** pos2, **unsigned int** n);

Satrga pos1 o'rindan boshlab, str satrining pos2 o'rnidan boshlab n ta belgini qo'shish.

insert(**unsigned int** pos1, **const char ***str, **int** n);

Satrga pos1 o'rindan boshlab, char toifasidagi satrning n ta belgisini qo'shish.

Satr qismini almashtirish. Satrning biror qismini almashtirish kerak bo'lsa, replace funksiyasidan foydalanish mumkin.

```

    replace (unsigned int pos1, unsigned int n1, const string &str);
    replace (unsigned int pos1, unsigned int n1, const string & str,
unsigned int pos2, unsigned int n2);
    replace (unsigned int pos1, unsigned int n1, const char *str,
int n);

```

replace funksiyasi insert kabi ishlaydi, faqat qo'shilishi kerak bo'lgan satrni pos1 - o'rindan boshlab n1 ta belgi o'rniga qo'shadi.

2 ta satrni to'la almashtirish uchun swap funksiyasi ishlatiladi.

Satrlarni solishtirish. Satrlarni solishtirish uchun compare funksiyasi ishlatiladi:

```

int compare (const string &str) const;
int compare (unsigned int pos1, unsigned int n1, const string
& str) const;
int compare (unsigned int pos1, unsigned int n1, const string
& str, unsigned int pos2, unsigned int n2) const;

```

compare funksiyasini chaqiruvchi satr, str o'zgaruvchisidagi satrdan kichik bo'lsa, manfiy qiymat qaytariladi. Katta bo'lsa musbat va teng bo'lsa 0 qiymat qaytariladi.

Satr qismini ajratib olish funksiyasi.

```

string substr(unsigned int pos=0, unsigned int n=npos) const;

```

Bu funksiya chaqiruvchi satrdan pos o'rnidan boshlab n belgini natija sifatida qaytaradi, agarda pos ko'rsatilmasa, satr boshidan boshlab n ta belgi ajratib olinadi.

```

string s,c;
s = "GULOY";
c = s.substr(3, 2) + s.substr(0, 3);
cout << c << endl; // OYGUL - chiqadi

```

Satrdan qidirish funksiyalari.

```

unsigned int find(const string &str, unsigned int pos=0) const;

```

Bu funksiyani chaqirgan satrning pos o'zgaruvchisida ko'rsatilgan joyidan boshlab str satrni qidiradi. Agar qidirilayotgan satr (str) topilsa, mos keluvchi satr qismining boshlanish indeksini javob sifatida qaytaradi, aks holda (satrning maksimal uzunligi qiymati) npos sonini qaytaradi. (npos=4294967295).

Agar pos ko'rsatilmasa, satr boshidan boshlab izlanadi.

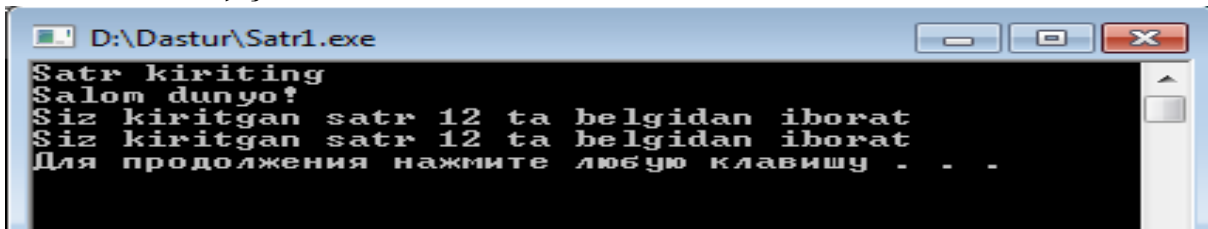
unsigned int rfind(**const** string &str, **unsigned int** pos=npos)
const;

Bu funksiyani chaqirgan satrdan pos o‘ringacha str satri qidiriladi. Agar str topilsa, oxirgi uchragan indeks qaytariladi.

Agar pos ko‘rsatilmasa, satr oxirigacha izlanadi. Ya’ni oxirgi uchragan indeks qaytariladi. Agar topilmasa, npos qaytariladi.

451. Satr uzunligini aniqlash

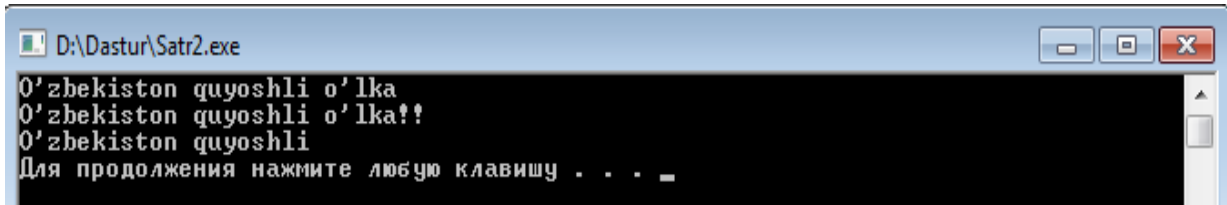
```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string s;
    cout << "Satr kiriting" << endl;
    getline(cin, s);
    cout<<"Siz kiritgan satr "<<s.length()
    <<" ta belgidan iborat";
    cout<<"Siz kiritgan satr "<<s.size()
    <<" ta belgidan iborat";
    return 0; }
```



452. Satr uzunligini o‘zgartirish

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    size_t n;
    string str ("I like to code in C");
    cout << str << endl;
    // satr uzunligini aniqlash
    n = str.size();
    // satr uzunligini 2 ta belgiga uzaytirish
    str.resize (n + 2, '+');
    cout << str << endl;
```

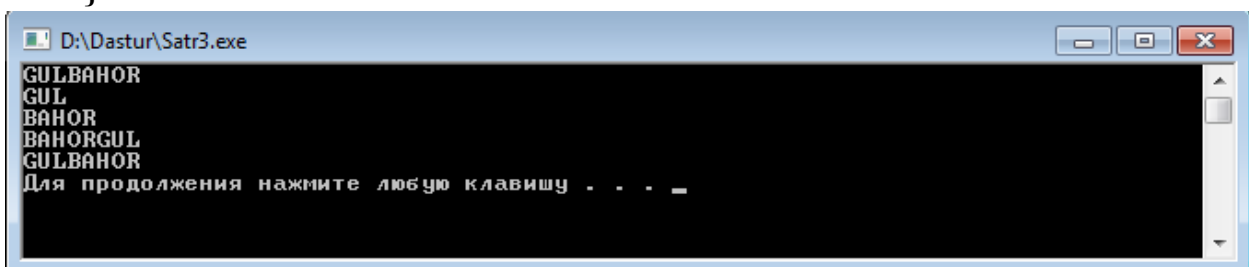
```
//satr uzunligini o'zgartirish
str.resize(14);
cout << str << endl;
return 0;
}
```



```
D:\Dastur\Satr2.exe
O'zbekiston quyoshli o'lka
O'zbekiston quyoshli o'lka!!
O'zbekiston quyoshli
Для продолжения нажмите любую клавишу . . . _
```

453. Satrdan nusxa olish

```
#include <iostream>
#include <string>
using namespace std;
int main() {
    string s1, s2, s3;
    s1 = "GULBAHOR";
    s2.assign(s1, 0, 3); // s2 = "GUL"
    s3.assign(s1, 3, 5); // s3 = "BAHOR"
    cout << s1 << endl;
    cout << s2 << endl;
    cout << s3 << endl;
    s1 = s3 + s2; // s1 = "BAHORGUL"
    cout << s1 << endl;
    s2.append(s3); // s2 = "GULBAHOR"
    cout << s2 << endl;
    return 0;
}
```



```
D:\Dastur\Satr3.exe
GULBAHOR
GUL
BAHOR
BAHORGUL
GULBAHOR
Для продолжения нажмите любую клавишу . . . _
```

454. char toifasidagi satrni string toifasiga o'girish.

```
#include <iostream>
#include <string.h>
using namespace std;
```

```

int main() {
char c[100];
string s = "Boshlang'ich qiymat";
s.clear(); // s ning qiymati tozalandi
cout << "Satr kiriting" << endl;
cin.getline(c, sizeof(c)); //s satriga char //toifasidagi c satrini
nusxasini olish
s.assign(c, strlen(c));
cout << s << endl;
return 0;
}

```

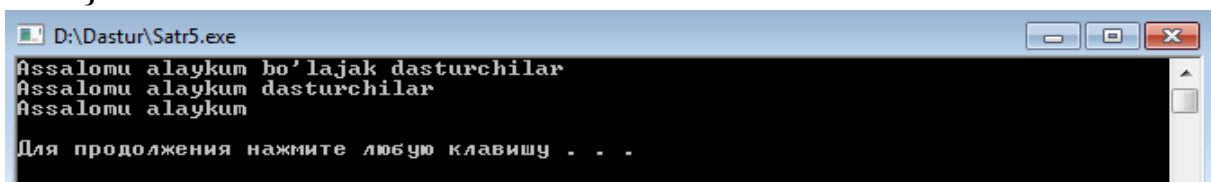


455. Satr qismini o'chirish

```

#include <iostream>
#include <string>
using namespace std;
int main() {
string s = "Assalomu alaykum bo'lajak dasturchilar";
cout << s << endl;
//16 - belgidan boshlab 9 ta belgini o'chirish s.erase(16, 9);
cout<<s<<endl;//Assalomu alaykum dasturchilar – //chiqadi
// s = "Assalomu alaykum dasturchilar";
// 16 - belgidan boshlab satr oxirigacha o'chirish
s.erase(16);
cout << s << endl; // Assalomu alaykum - chiqadi
s.erase();
cout << s << endl; // bo'sh satr chiqariladi
return 0;
}

```



456. Satrga boshqa satr qismini qo'shish

```

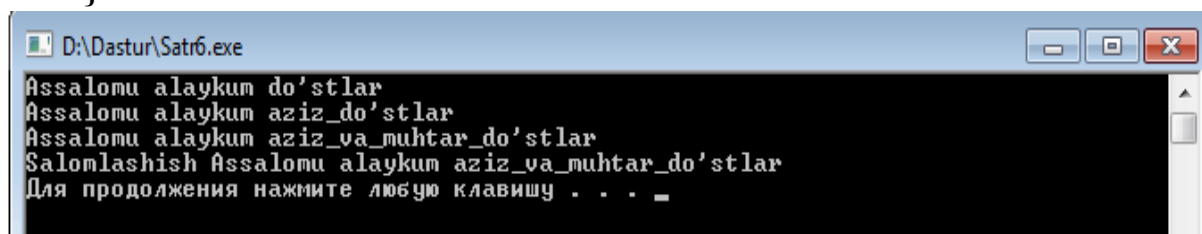
#include <iostream>

```

```

#include <string>
using namespace std;
int main() {
    string s = "Assalomu alaykum do'stlar";
    string c = "Merhibon_va_muhtarama_ayollar";
    char p[] = "Salomlashish odobi";
    cout << s << endl;
    // 17 - belgidan boshlab yangi satrni qo'shish
    s.insert(17, "aziz_");
    cout << s << endl; //Assalomu alaykum aziz do'stlar
    // c satridan 8 - belgidan boshlab 10 ta belgi
    // qo'shish
    s.insert(21, c, 8, 10);
    cout << s << endl; // Assalomu alaykum //
    aziz_va_muhtar_do'stlar
    // char toifasidagi satrdan 13 ta belgini qo'shish
    s.insert(0, p, 13);
    cout << s << endl;
    return 0;
}

```



457. Satr qismini almashtirish funsiyasiga misol

```

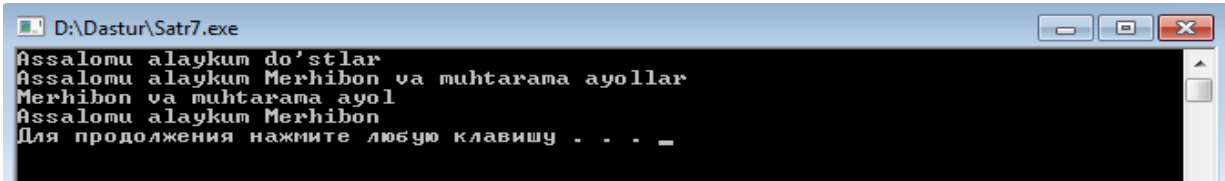
#include <iostream>
#include <string>
using namespace std;
int main() {
    string s = "Assalomu alaykum do'stlar";
    string c = "Merhibon va muhtarama ayol";
    cout << s << endl;
    // 17 - belgidan boshlab 5 ta belgi o'rniga c
    //satrni qo'shish
    s.replace(17, 5, c);
    cout << s << endl;
}

```

```

s.swap(c); // 2 ta satrni to'la almashtirish
cout << s << endl;
s.replace(0, 0, c, 0, 17);
s.erase(25);
cout << s << endl;
return 0;
}

```



```

D:\Dastur\Satr7.exe
Assalomu alaykum do'stlar
Assalomu alaykum Merhibon va muhtarama ayollar
Merhibon va muhtarama ayol
Assalomu alaykum Merhibon
Для продолжения нажмите любую клавишу . . .

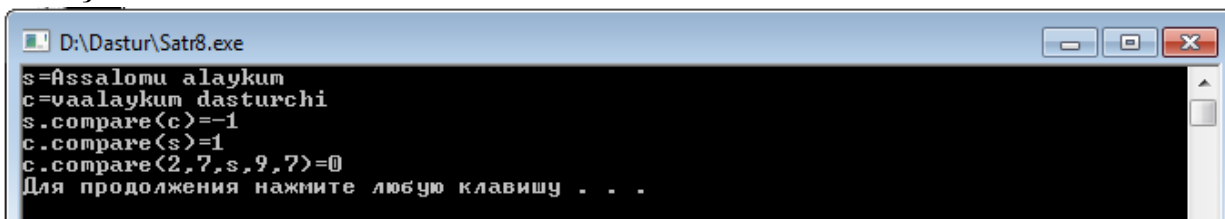
```

458. Satrlarni solishtirish.

```

#include <iostream>
#include <string>
using namespace std;
int main() {
string s = "Assalomu alaykum";
string c = "vaalaykum dasturchi";
cout << "s=" << s << endl;
cout << "c=" << c << endl;
cout << "s.compare(c)=" << s.compare(c) << endl; // -1
cout << "c.compare(s)=" << c.compare(s) << endl; // 1
cout << "c.compare(2,7,s,9,7)="
<< c.compare(2,7,s,9,7) << endl; // 0
return 0;
}

```



```

D:\Dastur\Satr8.exe
s=Assalomu alaykum
c=vaalaykum dasturchi
s.compare(c)=-1
c.compare(s)=1
c.compare(2,7,s,9,7)=0
Для продолжения нажмите любую клавишу . . .

```

459. Satrdan qidirish

```

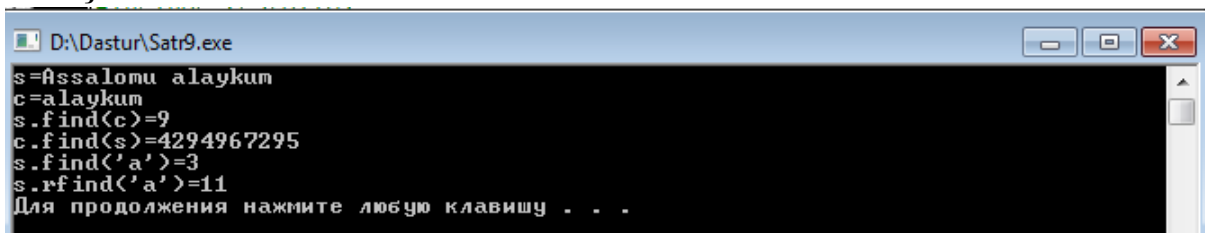
#include <iostream>
#include <string>
using namespace std;
int main() {
string s = "Assalomu alaykum";
string c = "alaykum";

```

```

cout << "s=" << s << endl;
cout << "c=" << c << endl;
cout << "s.find(c)=" << s.find(c) << endl; //9
cout << "c.find(s)=" << c.find(s) << endl;
// 4294967295
// birinchi uchragan "a" harfining o'rnini aniqlash
cout << "s.find('a')=" << s.find("a") << endl; // 3
// oxirgi uchragan "a" harfining o'rnini aniqlash
cout << "s.rfind('a')=" << s.rfind("a") << endl; // 11
return 0;
}

```



```

D:\Dastur\Satr9.exe
s=Assalomu alaykum
c=alaykum
s.find(c)=9
c.find(s)=4294967295
s.find('a')=3
s.rfind('a')=11
Для продолжения нажмите любую клавишу . . .

```

460. Satr qismini almashtirish

```

#include <iostream>
#include <stdlib.h>
#include <string>
using namespace std;
int main() {
    string c;
    string s, s1;
    size_t index;
    cout << "Satr kiriting" << endl;
    getline(cin, s);
    cout << "Qidiriladigan satrni kiriting=";
    getline(cin, c);
    cout << "Almashtiriladigan satrni kiriting=";
    getline(cin, s1);
    index = s.find(c);
    while (index < s.length()) {
        s.replace(index, c.length(), s1);
        index = s.find(c, index + s1.length());
    }
    cout << s << endl;
    system("pause");
}

```

```
return 0; }
```

Mustaqil ishlash uchun vazifa

461. Satr berilgan. Satrdagi raqamlar sonini aniqlovchi dastur tuzilsin.

462. Satr berilgan. Satrdagi katta lotin harflari sonini aniqlovchi dastur tuzilsin.

463. Satr berilgan. Satrdagi kichik lotin va kirill harflarining umumiy sonini aniqlovchi dastur tuzilsin.

464. Satr berilgan. Satrdagi xamma katta lotin harflari kichigiga almashtiruvchi dastur tuzilsin.

465. Satr berilgan. Satrdagi xamma katta harflarini kichigiga almashtiruvchi dastur tuzilsin. (lotinchani ham, kirillchani ham)

466. Satr berilgan. Satrdagi xamma katta harflarini kichigiga, kichiklarini kattasiga almashtiruvchi dastur tuzilsin (lotinchani ham, kirillchani ham).

467. Satr berilgan. Agar satrda butun son ifodalangan bo'lsa 1 chiqarilsin, agar haqiqiy son bo'lsa 2 chiqarilsin. Agar satrni songa aylantirish imkoni bo'lmasa 0 chiqarilsin. Haqiqiy sonning kasr qismi nuqta "." Bilan ajratilgan deb qabul qilinsin.

468. Butun musbat son berilgan. Uning belgilarini (raqamlarini) chapdan o'ngga qarab chiqaruvchi dastur tuzilsin.

469. Butun musbat son berilgan. Uning belgilarini (raqamlarini) ongdan chapga qarab chiqaruvchi dastur tuzilsin.

470. Butun musbat sonni ifodalovchi satr berilgan. Shu son raqamlari yig'indisini chiqaruvchi dastur tuzilsin.

471. "son $\pm$ son $\pm$... $\pm$ son" ko'rinishidagi arifmetik ifodani bildiruvchi satr berilgan. " $\pm$ " belgisi o'rnida yo "+" yoki "-" bo'ladi (Masalan: $7 + 3 - 2$). Ifodaning qiymatini aniqlovchi dastur tuzilsin (son o'rnida butun son kiritiladi).

472. Butun musbat sonni ifodalovchi ikkilik sanoq sistemasidagi son satrda berilgan. Bu sonning oʻnlik sanoq sistemasidagi qiymatin oʻzida saqllovchi satrni chiqaruvchi dastur tuzilsin.

473. Butun musbat sonni ifodalovchi oʻnlik sanoq sistemasidagi son satrda berilgan. Bu sonning ikkilik sanoq sistemasidagi qiymatin oʻzida saqllovchi satrni chiqaruvchi dastur tuzilsin.

474. N natural soni va satr berilgan. Uzunligi N ga teng boʻlgan satrni shunday hosil qilingki, agar kiritilgan satrning uzunligi N dan katta boʻlsa, satrning dastlabki belgilarini tashlab yuboring. Agar kiritilgan satrning uzunligi N dan kichik boʻlsa, yangi satr boshiga nuqtalarni “.” qoʻshing.

475. N_1 , N_2 natural sonlari va s_1 , s_2 satr berilgan. s_1 satrning dastlabki N_1 ta belgisidan va s_2 satrning oxirgi N_2 ta belgisidan iborat yangi satr hosil qiling.

476. C belgisi va S satri berilgan. S satrida uchragan har bir C belgisini 2 marta orttiruvchi dastur tuzilsin.

477. C belgisi va S_1 , S_2 satrlari berilgan. S_1 satriga shu satrda uchragan har bir C belgisidan oldin S_2 satrini qoʻshuvchi dastur tuzilsin.

478. C belgisi va S_1 , S_2 satrlari berilgan. S_1 satriga shu satrda uchragan har bir C belgisidan keyin S_2 satrini qoʻshuvchi dastur tuzilsin.

479. S_1 va S_2 satrlari berilgan. Agar S_2 satri S_1 satrida bor boʻlsa true, aks xolda false chiqaruvchi dastur tuzilsin.

480. S_1 va S_2 satrlari berilgan. S_2 satrini S_1 satrida takrorlanishlar sonini chiqaruvchi dastur tuzilsin.

481. S_1 va S_2 satrlari berilgan. S_1 satrida birinchi uchragan S_2 satrini oʻchirib tashlovchi dastur tuzilsin. Agar S_1 satrida S_2 satri uchramasa S_1 satri oʻzgarishsiz qoldirilsin.

482. S_1 va S_2 satrlari berilgan. S_1 satrida oxirgi uchragan S_2 satrini oʻchirib tashlovchi dastur tuzilsin. Agar S_1 satrida S_2 satri uchramasa S_1 satri oʻzgarishsiz qoldirilsin.

483. S_1 va S_2 satrlari berilgan. S_1 satrida uchragan barcha S_2 satrlarini oʻchirib tashlovchi dastur tuzilsin. Agar S_1 satrida S_2 satri uchramasa S_1 satri oʻzgarishsiz qoldirilsin.

484. S1, S2 va S3 satrlari berilgan. S1 satrida birinchi uchragan S2 satrini S3 satriga o'zgartiruvchi dastur tuzilsin.

485. S1, S2 va S3 satrlari berilgan. S1 satrida oxirgi uchragan S2 satrini S3 satriga o'zgartiruvchi dastur tuzilsin.

486. S1, S2 va S3 satrlari berilgan. S1 satridagi barcha S2 satrini S3 satriga o'zgartiruvchi dastur tuzilsin.

487. Kamida bitta probeldan iborat satr berilgan. Satridagi birinchi va ikkinchi probel orasidagi belgilarni chiqaruvchi dastur tuzilsin. Agar satr faqat bitta probeldan iborat bo'lsa, bo'sh satr chiqarilsin.

488. Kamida bitta probeldan iborat satr berilgan. Satridagi birinchi va oxirgi probel orasidagi belgilarni chiqaruvchi dastur tuzilsin. Agar satr faqat bitta probeldan iborat bo'lsa, bo'sh satr chiqarilsin.

489. Probel bilan ajratilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi so'zlar sonini aniqlovchi dastur tuzilsin.

490. Probel bilan ajratilgan va faqat katta harflar bilan terilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi bir xil harflar bilan boshlanuvchi va tugovchi so'zlar sonini aniqlovchi dastur tuzilsin.

491. Probel bilan ajratilgan va faqat katta harflar bilan terilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi kamida bitta 'A' harfi bor so'zlar sonini aniqlovchi dastur tuzilsin.

492. Probel bilan ajratilgan va faqat katta harflar bilan terilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi aniq 3 ta 'A' harfi bor so'zlar sonini aniqlovchi dastur tuzilsin.

493. Probel bilan ajratilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi eng qisqa so'z uzunligini aniqlovchi dastur tuzilsin.

494. Probel bilan ajratilgan o'zbekcha so'zlardan iborat satr berilgan. Satrdagi eng uzun so'z uzunligini aniqlovchi dastur tuzilsin.

495. Probel bilan ajratilgan o'zbekcha so'zlardan iborat satr berilgan. Shu satrdagi so'zlarni "." bilan ajratilgan xolda chiqaruvchi dastur tuzilsin. Satr oxiriga "." qo'yish shart emas.

496. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. Shu satrdagi so‘zlarni quyidagicha o‘zgartiring: Har bir so‘zning birinchi harfi bilan bir xil bo‘lsan harflarni “.” bilan almashtiruvchi dastur tuzilsin. Masalan: “minimum” so‘zi “mini.u.” bo‘lib o‘zgartiriladi. Probellar soni o‘zgarishsiz qolsin.

497. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. Shu satrdagi so‘zlarni quyidagicha o‘zgartiring: Har bir so‘zning oxirgi harfi bilan bir xil bo‘lsan harflarni “.” bilan almashtiruvchi dastur tuzilsin. Masalan: “minimum” so‘zi “.ini.um” bo‘lib o‘zgartiriladi. Probellar soni o‘zgarishsiz qolsin.

498. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. So‘zlar orasida probel bir nechta bo‘lishi mumkin. Shu satrdagi ortiqcha probellarni olib tashlovchi va satrni teskari tartibda chiqaruvchi dastur tuzilsin.

499. Probel bilan ajratilgan va katta harflar bilan terilgan o‘zbekcha so‘zlardan iborat satr berilgan(ketma – ket bir nechta probel bo‘lishi mumkin). Satrdagi so‘zlarni alfavit tartibida bitta probel bilan ajratilgan xolda chiqaruvchi dastur tuzilsin.

500. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. Satrdagi har bir so‘zning birinchi harfini kattasi bilan almashtiruvchi dastur tuzilsin. So‘z deganda probel, satr boshi yoki satr oxiri bilan ajratilgan belgilar ketma – ketligi tushuniladi.

501. Satr berilgan. Satrdagi tinish belgilari sonini chiqaruvchi dastur tuzilsin.

502. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. Satrdagi katta harflar sonini chiqaruvchi dastur tuzilsin.

503. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. Satrdagi eng uzun so‘zni chiqaruvchi dastur tuzilsin. Agar eng uzunlari bir nechta bo‘lsa, faqat birinchisi chiqarilsin.

504. Probel bilan ajratilgan o‘zbekcha so‘zlardan iborat satr berilgan. Satrdagi eng qisqa so‘zni chiqaruvchi dastur tuzilsin. Agar eng qisqalari bir nechta bo‘lsa, faqat oxirgisi chiqarilsin.

505. Satr berilgan. Satrdagi ortiqcha probellarni olib tashlovchi dastur tuzilsin.

FAYLLAR BILAN ISHLASH

Ma'lumotlarni saqlab qo'yish uchun, tashqi xotiraning nomlangan qismiga fayl deyiladi. Bunday fayllar fizik fayllar deyiladi.

Mantiqiy fayllar. Fizik fayllar bilan ishlash uchun, dasturlashtirish tillarida maxsus strukturalashgan, toifalangan fayllar kiritilgan. Bunday fayllar mantiqiy fayllar deyiladi. Mantiqiy fayllar, hech qanday fizik xotirani band qilmasdan ma'lumotlarning mantiqiy modelini o'zida saqlaydi. Fizik va mantiqiy fayllar bir - biri bilan fopen funksiyasi orqali bog'lanadi. Fayl bir nechta elementdan tashkil topgan bo'lganligi uchun, faqat fayl ko'rsatkichi ko'rsatayotgan elementga murojaat qilish mumkin.

Fayldan o'qish yoki yozish mumkin bo'lgan o'rinni ko'rsatish elementga fayl ko'rsatkichi deyiladi. Fayldan ma'lumot o'qiganda yoki yozganda fayl ko'rsatkichi avtomat ravishda o'qilgan yoki yozilgan bayt miqdoricha siljiydi. Fayl ko'rsatkichini magnitafon galovkasiga o'xshatish mumkin.

Binar fayl - har xil ob'yektlarni ifodalovchi baytlar ketma - ketligidir. Ob'yektlar faylda qanday ketma - ketlikda joylashganini dasturning o'zi aniqlashi lozim. Fayllar bilan ishlovchi funksiyalardan foydalanish uchun <stdio.h> sarlavha faylini dasturga qo'shish kerak bo'ladi.

Fayldan ma'lumotlarni o'qish yoki yozish uchun ochish fopen funksiyasi orqali amalga oshiriladi.

FILE* fopen(**const char** *filename, **const char*** mode);

filename - o'zgaruvchisi **char** toifasidagi satr bo'lib, faylning to'liq nomini ko'rsatishi lozim (filename = "D:\dastur\fayl\Fayl.txt"). Agar faylning faqat nomi ko'rsatilgan bo'lsa, fayl joriy katalogdan qidiriladi (filename = "fayl.txt").

mode - o'zgaruvchisi ham **char** toifasidagi satr bo'lib, faylni qaysi xolatda ochish lozimligini bildiradi. mode qiymati faylning ochilish xolati:

"w" faylni yozish uchun ochish. filename o'zgaruvchisida ko'rsatilgan fayl hosil qilinadi va unga ma'lumot yozish mumkin bo'ladi. Agar fayl oldindan bor bo'lsa (ya'ni oldin hosil qilingan

bo'lsa), faylning ma'lumotlari o'chiriladi va yangi bo'sh fayl faqat yozish uchun ochiq holda bo'ladi.

"r" fayl o'qish uchun ochiladi. Agar fayl oldindan mavjud bo'lmasa, xatolik sodir bo'ladi. Ya'ni ochilishi lozim bo'lgan fayl oldindan hosil qilingan bo'lishi shart.

"a" faylga yangi ma'lumotlar qo'shish - kiritish uchun ochiladi. Yangi kiritilgan ma'lumotlar fayl oxiriga qo'shiladi. Agar fayl oldindan mavjud bo'lmasa, yangi fayl hosil qilinadi.

"w+" yozish va o'qish uchun faylni ochish. Agar fayl oldindan bor bo'lsa (ya'ni oldin hosil qilingan bo'lsa), faylning ma'lumotlari o'chiriladi va yangi bo'sh fayl yozish va o'qish uchun ochiq holda bo'ladi.

"r+" oldindan mavjud bo'lgan faylni o'qish va yozish uchun ochish.

"a+" fayl ma'lumotlarni o'qish va yangi ma'lumot qo'shish uchun ochiladi. fseek, rewind faylni ochishda xatolik sodir bo'lsa, fopen funksiyasi NULL qiymat qaytaradi.

Ochilgan faylni yopish uchun fclose funksiyasi ishlatiladi.

int fclose (FILE * stream);

Faylga ma'lumot yozish va o'qish.

size_t fread(**void** *ptr, size_t size, size_t n, FILE *stream);

fread funksiyasi, fayldan ptr ko'rsatkichi adresiga size xajmdagi ma'lumotdan n tani o'qishni amalga oshiradi. Agar o'qish muvoffaqiyatli amalga oshsa fread funksiyasi o'qilgan bloklar soni n ni qaytaradi. Aksholda nol qaytariladi:

size_t fwrite (**const void** *ptr, size_t size, size_t n, FILE *stream);

fwrite funksiyasi, faylga ptr ko'rsatkichi adresidan boshlab size xajmdagi ma'lumotdan n tani yozishni amalga oshiradi.

Satrnı yozish va o'qish uchun quyidagicha kod ishlatiladi:

fwrite(s, sizeof(**char**), strlen(s) + 1, f);

fread(s, sizeof(**char**), strlen(s) + 1, f);

buning kamchiligi s satridagi har bir belgi alohida - alohida faylga yozildi va o'qildi.

Bu masalani quyidagicha hal qilish mumkin edi:

fwrite(s, sizeof(s), 1, f);

fread (s, sizeof(s), 1, f);

lekin bu usulning ham kamchiligi bor. Ya'ni s satri belgilar i soni massiv o'lchamidan kam bo'lgan holda, keraksiz ma'lumotlarni saqlash va o'qish sodir bo'ladi.

Fayl ko'rsatkichi bilan ishlovchi funksiyalar. Fayldan ma'lumot o'qiganda yoki yozganda fayl ko'rsatkichi avtomat ravishda o'qilgan yoki yozilgan bayt miqdoricha siljiydi. Fayl ko'rsatkichining kelgan joyini aniqlash uchun *ftell* funksiyasi ishlatiladi.

long int ftell (FILE * stream);

Fayl ko'rsatkichini siljitish uchun fseek funksiyasi ishlatiladi.

int fseek (FILE *stream, **long int** offset, **int** whence);

Bu funksiya offset da ko'rsatilgan bayt miqdoricha siljishni amalga oshiradi. whence o'zgaruvchisi quyidagi qiymatlarni qabul qilishi mumkin:

O'zgarmas whence Izoh SEEK_SET 0 Fayl boshiga nisbatan siljitish, SEEK_CUR 1 Fayl ko'rsatkichining joriy xolatiga nisbatan siljitish, SEEK_END 2 Fayl oxiriga nisbatan siljitish.

Agar whence = 1 bo'lsa (SEEK_CUR), offset musbat (o'ngga siljish) yoki manfiy (chapga siljish) bo'lishi mumkin.

Fayl ko'rsatkichini faylning boshiga o'rnatish uchun rewind funksiyasi ishlatiladi.

void rewind (FILE *stream);

Bu amalni fayl ko'rsatkichini siljitish orqali ham amalga oshirish mumkin.

fseek (f, 0, SEEK_SET);

Agar faylda faqat butun sonlar yozilgan bo'lsa, uning k - elementiga murojaat quyidagicha bo'ladi.

fseek (f, sizeof(**int**) * (k - 1), SEEK_SET);

fread (&n, sizeof(**int**), 1, f);

Fayl oxirini aniqlash uchun feof funksiyasi ishlatiladi.

int feof (FILE * stream);

feof funksiyasi fayl ko'rsatkichi fayl oxirida bo'lsa, noldan farqli qiymat qaytaradi. Qolgan hollarda nol qaytaradi.

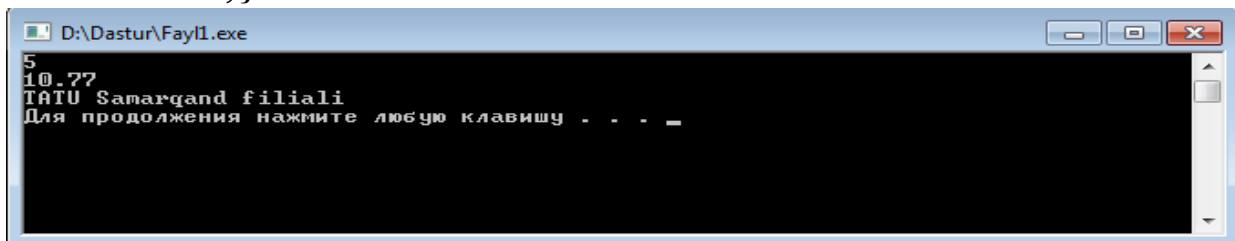
506. fread va fwrite funksiyalarining qo'llanilishi

#include <cstdlib>

```

#include <iostream>
#include <string.h>
#include <stdio.h>
using namespace std;
int main() {
int n = 5;
double d = 10.77;
char s[30] = "TATU Samarqand filiali";
FILE *f;
// binar faylni yozish uchun ochamiz
f = fopen("fayl1.dat", "wb");
fwrite(&n,sizeof(int),1,f);//n sonini faylga yozish
fwrite(&d,sizeof(double),1,f);//d sonini faylga //yozish
// satrni faylga yozish
fwrite(s,sizeof(char),strlen(s)+1,f);
fclose(f); // faylni yopish
n = 0; d = 0;
// binar faylni o'qish uchun ochamiz
f = fopen("fayl1.dat", "rb");
fread(&n,sizeof(int),1,f);//n sonini fayldan o'qish
fread(&d,sizeof(double),1,f);//d sonini fayldan //o'qish
// satrni fayldan o'qish
fread(s, sizeof(char), strlen(s) + 1, f);
fclose(f); // faylni yopish
cout << n << endl;
cout << d << endl;
cout << s << endl;
return 0;}

```



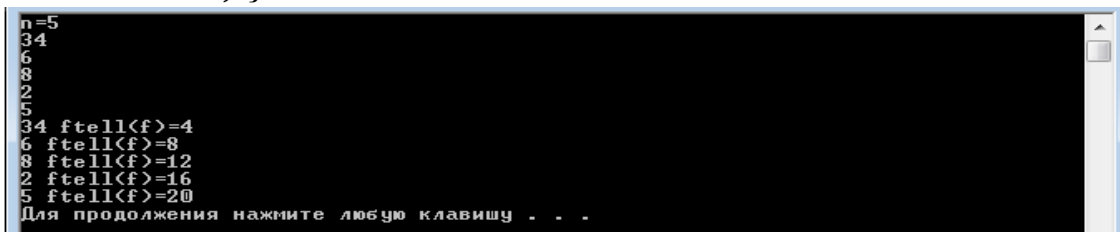
507. n natural soni berilgan. Elementlari n ta butun sonidan iborat bo'lgan faylni hosil qiluvchi va ekranga chiqaruvchi dastur tuzilsin.

```
#include <cstdlib>
```

```

#include <iostream>
#include <stdio.h>
using namespace std;
int main() {
int n, k;
FILE *f;
f = fopen("binar", "wb+");
// binar faylni yozish va o'qish uchun ochish
if (f == NULL) {
cout << "Faylni hosil qilishda xato bo'ldi";
return 1;
}
cout << "n="; cin >> n;
for (int i = 0; i < n; i++) {
cin >> k;
fwrite(&k, sizeof(k), 1, f); }
// fayl ko'rsatkichini fayl boshiga qo'yish
rewind(f);
while (fread(&k, sizeof(k), 1, f)) {
//fayl boshidan fayl ko'rsatkichi turgan //o'ringacha bo'lgan
baytlar
int bayt = ftell(f);
cout << k << " ftell(f)=" << bayt << endl;
}
fclose(f);
return 0; }

```



```

n=5
34
6
8
2
5
34 ftell(f)=4
6 ftell(f)=8
8 ftell(f)=12
2 ftell(f)=16
5 ftell(f)=20
Для продолжения нажмите любую клавишу . . .

```

508. n natural soni berilgan. Elementlari n ta butun sondan iborat bo'lgan faylni hosil qiluvchi va juft elementlarini 2 marta orttiruvchi dastur tuzilsin.

```

#include <cstdlib>
#include <iostream>
#include <stdio.h>

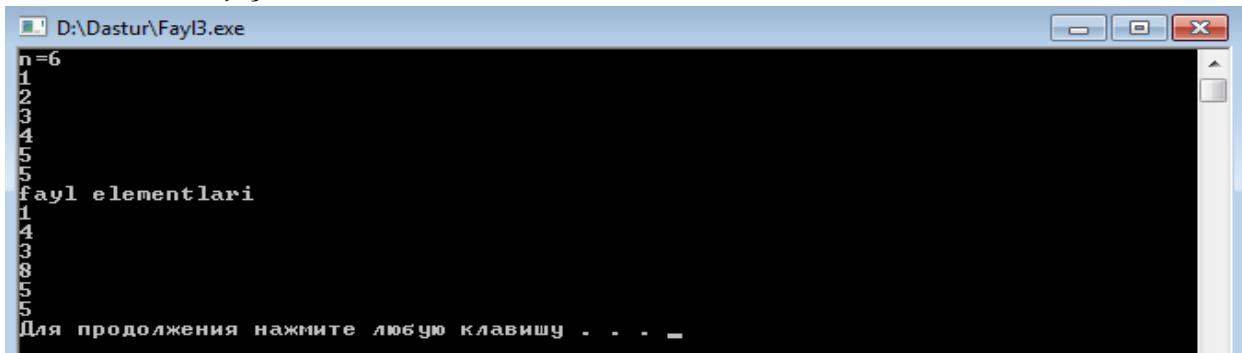
```

```

using namespace std;
int main() {
int n, k;
FILE *f;
f = fopen("binar", "wb+");
cout << "n="; cin >> n;
for (int i = 0; i < n; i++){
cin >> k;
fwrite(&k, sizeof(k), 1, f); }
// fayl ko'rsatkichini fayl boshiga qo'yish
rewind(f);
while (!feof(f)) {
// fayl ko'rsatkichi o'rnini eslab qolish
int pos = ftell(f);
fread(&k, sizeof(k), 1, f);
if (k % 2 == 0 ) {
k *= 2;
//fayl ko'rsatkichini oldingi xolatiga
//o'rnatish
fseek(f, pos, SEEK_SET);
fwrite(&k, sizeof(int), 1, f);
//fayl ko'rsatkichi o'rnini sizeof(int) ga
//surish
pos += sizeof(int);
fseek(f, pos, SEEK_SET); } }
cout << "fayl elementlari\n";
rewind(f);
// fayl elementlarini chiqarish
while (fread(&k, sizeof(k), 1, f))
cout << k << endl;
fclose(f);

```

```
return 0; }
```



```
D:\Dastur\Fayl3.exe
n=6
1
2
3
4
5
5
fayl elementlari
1
4
3
8
5
5
Для продолжения нажмите любую клавишу . . . _
```

Mustaqil ishlash uchun vazifa

509. Oilangizning har bir a'zosining boyliklari haqida ma'lumotlardan tashkil topgan turlashgan faylni yarating.

510. N ta haqiqiy sonlar ketma-ketligini faylga yozing. Fayl komponentalarining kupaytmasini hisoblang va ekranga chiqaring.

511. Tasodifiy sonlar generatori yordamida hosil qilingan natural sonlar bilan f faylni to'ldiring. Komponentalari orasidan toq sonlar kvadratlari sonini toping.

512. Elementlari alohida so'z hisoblanuvchi turlashgan fayl berilgan. "t" harfi bilan boshlanuvchi hamma so'zlarni chop qiling. Bunda ikki variantni qarang:

- a) ma'lumki, mavjud faylga 30 ta so'z yozilgan;
- b) mavjud fayl o'lchami ma'lum emas.

513. Elementlari sonli bo'lgan mavjud turlashgan faylni berilgan son bilan almashtiring (yangi qiymat klaviaturadan kiritiladi):

- a) birinchi elementni;
- b) beshinchi elementni;
- c) k-nchi elementni;
- d) oxirgi elementni.

514. Elementlari alohida so'zlar bo'lgan mavjud turlashgan faylni berilgan qiymat bilan almashtiring (yangi so'z klaviaturadan kiritiladi):

- a) birinchi elementni;
- b) uchinchi elementni;
- c) s-nchi elementni;
- d) oxirgi elementni.

515. Turlashgan fayl elementlari alohida soʻz hisoblanadi. Uning barcha elementlarini boshqa soʻz bilan almashtiring. Bunda ikkita variantni qarang:

- a) maʼlumki, mavjud faylga 12 ta soʻz yozilgan;
- b) mavjud fayl hajmi maʼlum emas.

516. Sonli turlashgan fayl mavjud. Uning tartib nomeri 3 ga karrali boʻlgan hamma elementlarini almashtiring. Yangi qiymat klaviaturadan kiritiladi. Bunda ikkita variantni qarang:

- a) maʼlumki, mavjud faylga 20 ta son yozilgan;
- b) mavjud fayl hajmi maʼlum emas.

517. Butun sonli turlashgan fayl mavjud. Uning hamma juft elementlarini nol bilan almashtiring. Bunda ikkita variantni qarang:

- a) maʼlumki, mavjud faylga 13 ta son yozilgan;
- b) mavjud fayl hajmi maʼlum emas.

518. Elementlari alohida soʻzlardan iborat turlashgan fayl mavjud. Agar soʻz “k” harfi bilan boshlansa, uni “K” harfi bilan almashtiring.

519. Sonli turlashgan fayl mavjud. Toping (hamma holatlarda fayl hajmi maʼlum emas):

- a) faylning birinchi va ikkinchi sonlar yigʻindisini;
- b) faylning k–nchi va q–nchi sonlari yigʻindisi;
- c) faylning birinchi va oxirgi sonlari kupaytmasini;
- d) faylning hamma sonlarining yigʻindisini;
- e) fayldagi sonlar migʻdorini;
- f) a sonidan oshib ketmaydigan fayldagi sonlar migʻdorini;
- g) faylning musbat sonlarining oʻrta arifmetigini;
- h) mavjud fayldagi maksimal sonni;
- i) fayldagi minimal sonning tartib nomeri (agar bunday fayllar bir nechta boʻlsa, ularning birinchisini nomerini toping).

520. Elementlari alohida soʻzlardan iborat turlashgan fayl mavjud. Toping (hamma holatlarda fayl hajmi maʼlum emas):

- a) “m” harfi bilan boshlanuvchi soʻzlar sonini;
- b) “l” harfi bilan boshlanuvchi barcha soʻzlar va ularning birinchisini tartib nomerini chop qiling;
- c) tartib nomeri juft boʻlgan soʻzlar va ulardan soʻz tuzing;
- d) eng uzun soʻzni;

e) eng qisqa soʻzni.

521. Turlashgan faylga shahar nom iva uning aholisi soni yozilgan. Har bir shaharning aholisini 5% ga oshiring (aholi migʻdori – har doim butun son).

522. Butun sonli turlashgan fayl mavjud. Undan birinchi noldan keyin yozilgan sonni oʻchiring (faylda nol mavjud deb qabul qilinsin). Natija boshqa faylga yozilsin.

523. Elementlari alohida belgilardan iborat turlashgan fayl mavjud. Undan birinchi “o” harfini oʻchiring (“o” harfi faylda mavjud deb qabul qilinsin). Natija boshqa faylga yozilsin.

524. Butun sonli turlashgan fayl mavjud. Birinchi 100 sonidan keyin 100 sonini qoʻying. Natija boshqa faylga yozilsin.

525. Oʻttizta sonli turlashgan fayl mavjud. Boshqa faylga mavjud faylni teskari tartibda yozilsin.

526. Elementlari 20 ta alohida belgidan iborat turlashgan fayl mavjud. Boshqa faylga mavjud fayl belgilarini teskari tartibda yozilsin.

527. Elementlari sondan iborat, oʻlchamlari bir xil turlashgan ikkita fayl mavjud. Dastlab birinchi faylning, keyin ikkinchisini sonlarini yozib uchinchi faylni hosil qiling. Sonlar ketma-ketlik tartibi saqlansin.

528. Elementlari alohida belgilardan iborat turlashgan fayl mavjud. Bu faylning barcha raqamlarini ikkinchi faylga, qolgan belgilarni esa uchinchi faylga yozing. Sonlar ketma-ketlik tartibi saqlansin.

529. Elementlari alohida belgilardan iborat boʻlgan, oʻlchamlari bir xil turlashgan ikkita fayl mavjud. Ularning elementlarini mos kelishini aniqlang. Agar mos kelmasa, u holda bu fayllarning bir-biridan farq qiluvchi birinchi komponenta nomerini oling.

MATNLI FAYLLAR

Matnli fayllar bilan ishlash binar fayllar bilan ishlashdan bir oz farq qiladi. Matnli fayllarda ma'lumotlar satrlarda saqlanadi. Matnli fayl elementlari har xil uzunlikdagi satrlardir. Bu satrlar bir biridan satr oxiri belgisi bilan ajratiladi. Matnli fayl elementlari indekslanmagan bo'lganligi uchun, faylning istalgan elementiga bevosita murojaat qilib bo'lmaydi.

C++ da matnli yoki binar fayllar bilan ishlash uchun keng imkoniyatlar berilgan. Matnli fayllar bilan ishlashda oddiy C ning funksiyalaridan ham foydalanish mumkin.

Masalan, formatli o'qish va yozish funksiyalari yoki oldingi mavzudagi funksiyalardan foydalanishimiz mumkin. Matnli fayllar bilan ishlashning bunday usuli kitoblarda keng yoritilgan. Ularni mustaqil o'qib - o'rganishingiz mumkin.

Bu mavzu fayllar bilan ishlovchi oqimlarni qisqacha o'rganamiz va buni matnli fayl misolida ko'ramiz.

Fayllar bilan ishlash uchun quyidagi sifnlar ob'yektlari hosil qilinadi:

- ofstream - faylga ma'lumot yozish uchun
- ifstream - fayldan ma'lumot o'qish uchun
- fstream - fayldan ma'lumot o'qish uchun va yozish uchun

Bu sinflarni dasturda ishlatish uchun <fstream> sarlavha faylini qo'shish kerak bo'ladi. Bundan keyin dasturda aniq fayllar oqimini aniqlash mumkin. Masalan:

ofstream yozish;//faylga yozish oqimini e'lon qilish

ifstream oqish;//fayldan o'qish oqimini e'lon qilish

fstream yoz_oqi;//faylga yozish va o'qish oqimini //e'lon qilish

Keyin faylni ochish kerak bo'ladi. Faylni ochish deganda, uning ustida nima amal qilinishi haqida amaliyot tizimiga xabar berish tushuniladi.

void open (**const char** *filename, ios_base::openmode mode = ios_base::out);

mode parametri quyidagicha qiymatlarni qabul qilishi mumkin:

ios::in faqat ma'lumot o'qish uchun

ios::out faqat ma'lumot yozish uchun

ios::ate faylni ochishda fayl ko'rsatkichini fayl oxiriga qo'yish

ios::app fayl oxiriga ma'lumotlarni yozish uchun
ios::trunc bor bo'lgan faylning ustidan yangi faylni yozish
ios::binary binar holda ma'lumotlarni almashish uchun

Har bir sinf uchun mode parametrining odatiy qiymatlari mavjud:

class default mode parameter

ofstream ios::out

ifstream ios::in

fstream ios::in | ios::out

Fayl ustida o'qish yoki yozish amalini bajarib bo'lgandan song, faylni yopish kerak bo'ladi. Faylni yopish uchun close funksiyasi ishlatiladi. Masalan:

yozish.close();

oqish.close();

istream sinfi funksiyalari:

istream& seekg(streampos pos);

istream& seekg(streamoff off,ios_base::seekdir dir);

o'qish oqimi ko'rsatkichini o'rnatish (siljitish).

pos - oqim buferining yangi pozitsiyasi.

dir parametri quyidagilardan birini qabul qilishi mumkin:

Qiymat	Izoh
ios::beg	oqimning boshlanishi
ios::cur	oqimning joriy xolari
ios::end	oqim oxiri

long tellg(); o'qish oqimining joriy xolatini (pozitsiyasi) aniqlash.

ostream sinfi funksiyalari:

ostream& seekp(streampos pos);

ostream& seekp(streamoff off,ios_base::seekdir dir);

Yozish oqimi o'rni (pozitsiyasini) o'rnatish.

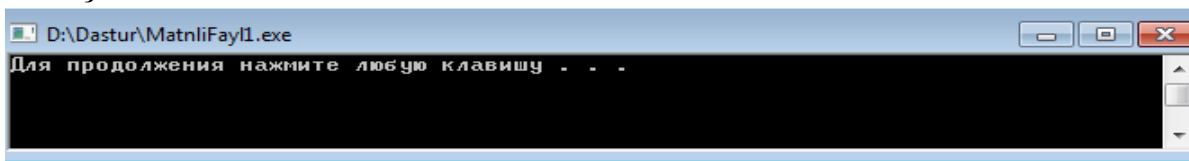
pos - oqim buferining yangi pozitsiyasi.

dir parametri beg, cur, end qiymatlaridan birini qabul qilishi mumkin.

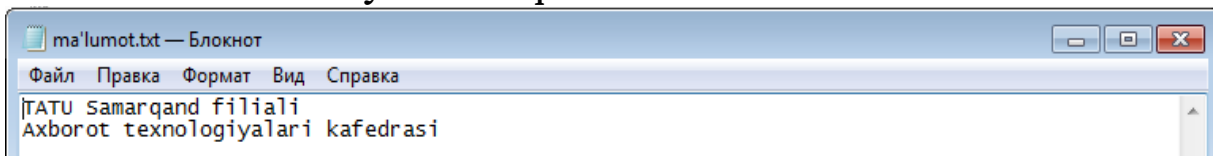
long tellp() - yozish oqimining kelgan joyini aniqlash.

530. Matnli faylga ma'lumot yozish

```
#include <cstdlib>
#include <iostream>
#include <fstream>
using namespace std;
int main(){
    ofstream yozish; //faylga yozish oqimini hosil //qilish
    yozish.open("D:/Dastur/File/ma'lumot.txt");
    // yangi ma'lumot.txt nomli fayl hosil qilinadi.
    // agar ma'lumot.txt fayli oldindan bo'lsa,
    // uning eski qiymatlari o'chiriladi
    // va yangi fayl hosil qilinadi
    yozish<<"TATU Samarqand filiali"<<endl;
    yozish<<"Axborot texnologiyalari kafedrası"<<endl;
    yozish.close(); // faylni yopish
    return 0;
}
```



ma'lumot.txt fayli hosil qilinadi.



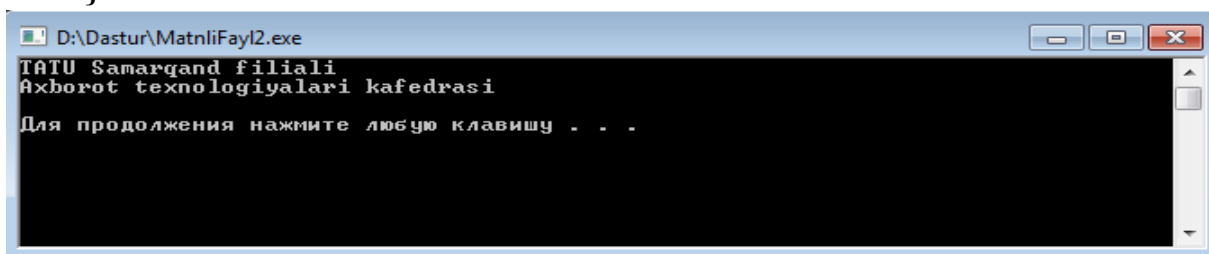
531. Matnli fayldan o'qish

```
#include <cstdlib>
#include <iostream>
#include <fstream>
using namespace std;
int main() {
    ifstream oqish; //fayldan o'qish oqimini hosil //qilish
    string satr;
    oqish.open("D:/Dastur/File/ma'lumot.txt");
    // faylni ochishda xatolik sodir bo'lsa
```

```

if (!oqish.is_open()){
cout<<"Faylni ochishda xatolik sodir bo'ldi.";
exit(1); // dasturni tugatish }
while (!oqish.eof()){
// fayldan o'qish
getline(oqish, satr);
// ekranga chiqarish
cout << satr << endl; }
//ma'lumot.txt fayli bilan oqish oqimi aloqasini
//uzish
oqish.close();
return 0;
}

```



532. Fayldan nusxa olish.

```

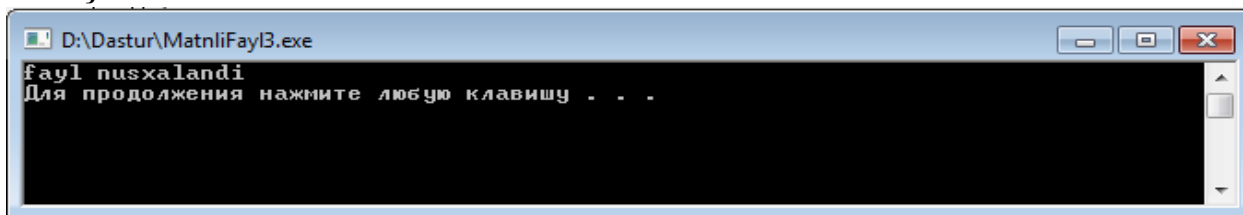
#include <cstdlib>
#include <iostream>
#include <fstream>
using namespace std;
int main(int argc, char *argv[]) {
int length;
char *buffer, fayl[] = "D:/Dastur/File/matn.txt";
yangi[]="D:/Dastur/File/yangi_fayl.txt";
// fayl - nusxalanadigan fayl nomi
// yangi - yangi nusxalangan fayl nomi
// o'qish oqimi
ifstream fromfile(fayl, ios::binary);
if (!fromfile.is_open()) {
cout<<"faylni o'qishda xatolik sodir bo'ldi\n";
exit(1);
}
// yozish oqimi
ofstream tofile(yangi, ios::binary );

```

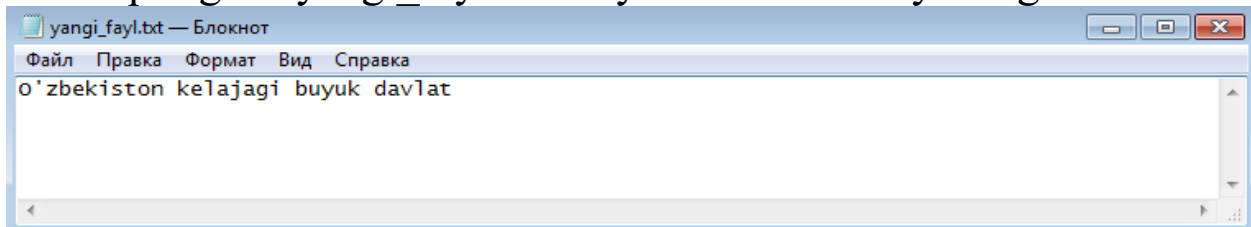
```

// fayl xajmini aniqlash:
fromfile.seekg(0,ios::end); //fayl oxiriga o'tish
length = fromfile.tellg();
fromfile.seekg(0,ios::beg); //fayl boshiga o'tish
// xotira ajratish:
buffer = new char [length];
// blokka ma'lumotlarni o'qish:
fromfile.read (buffer, length);
fromfile.close();
// nusxalanishi kerak bo'lgan faylga yozish
tofile.write (buffer, length);
// xotirani bo'shatish
delete[] buffer;
cout << "fayl nusxalandi\n";
return 0;
}

```



Hosil qilingan “yangi_fayl.txt” fayli “matn.txt” faylining nusxasi.



533. lug‘at.txt nomli fayl berilgan. Faylning har bir satrida inglizcha va o‘zbekcha so‘zlar “->” belgisi bilan ajratilgan. Inglizcha so‘zlarni inglizcha so‘zlar.txt fayliga, o‘zbekcha so‘zlarni o‘zbekcha so‘zlar.txt fayliga yozuvchi dastur tuzilsin.

lug‘at.txt fayli quyidagicha bo‘ladi:

hello->salom

good->yahshi

bad->yomon

you->sen

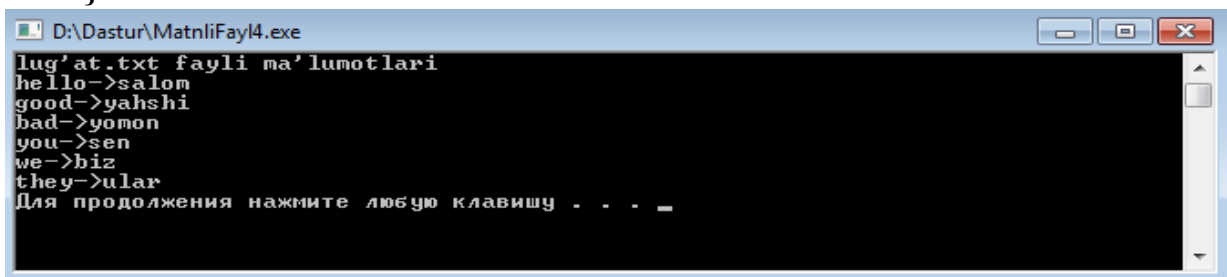
we->biz

they->ular

```

#include <cstdlib>
#include <iostream>
#include <fstream>
#include <string>
using namespace std;
int main() {
    ifstream dic("D://Dastur//File//lug‘at.txt");
    ofstream uzbek("D://Dastur//File//o‘zbekcha so‘zlar.txt");
    ofstream english("D://Dastur//File//inglizcha so‘zlar.txt");
    if (!dic.is_open()) {
        cout << "lug‘at.txt - fayli topilmadi\n";
        exit(1);
    }
    string s, uzb, eng;
    int p;
    cout << "lug‘at.txt fayli ma'lumotlari\n";
    while (!dic.eof()) {
        getline(dic, s);
        p = s.find("->");
        eng.assign(s, 0, p - 1);
        uzb.assign(s, p + 2, s.length() - (p + 2));
        uzbek << uzb << endl;
        english << eng << endl;
        cout << s << endl;
    }
    dic.close();
    uzbek.close();
    english.close();
    return 0;
}

```

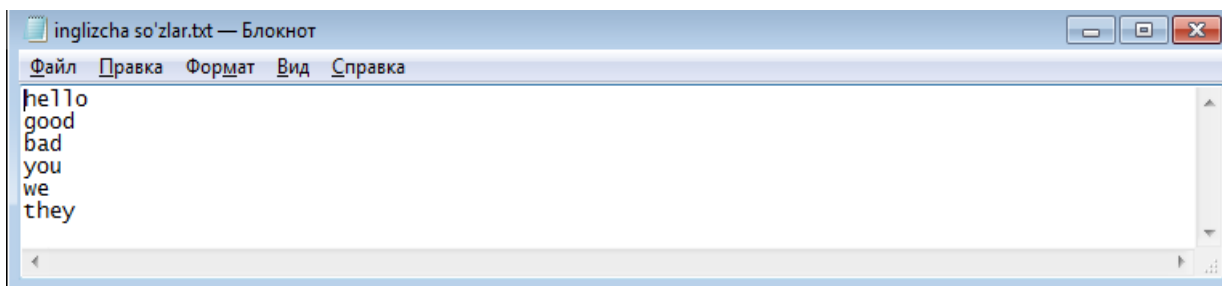


```

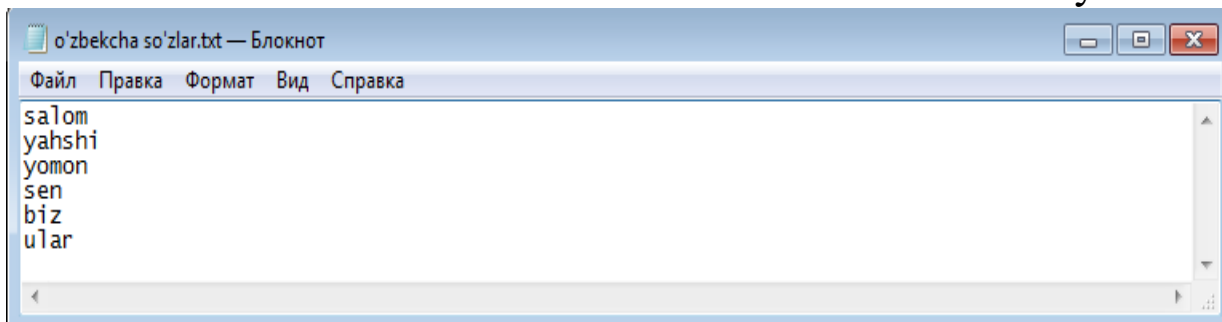
D:\Dastur\MatnliFayl4.exe
lug‘at.txt fayli ma'lumotlari
hello->salom
good->yahshi
bad->yomon
you->sen
we->biz
they->ular
Для продолжения нажмите любую клавишу . . .

```

Inglizcha so‘zlardan iborat “inglizcha so‘zlar.txt” fayli:



O'zbekcha so'zlardan iborat "o'zbekcha so'zlar.txt" fayli:



Mustaqil ishlash uchun vazifa

534. Tashqi matnli fayl mavjud. Uning satrlaridan birinchi eng qisqasini chop qiling.

535. T matnli faylga bo'sh bo'lmagan haqiqiy sonlar ketma-ketligi bo'sh joy bilan ajratilib yozilgan. Bu sonlarning eng kattasini topish uchun $\text{Max}(T)$ funksiyasini tavsiflang.

536. Matnli fayl mavjud. Undan o'chiring:

- a) uchinchi satrni;
 - b) oxirida so'roq belgisi turgan birinchi satrni.
- Ikkala holatda ham natija boshqa faylga yozilsin.

537. Matnli fayl mavjud. Chop qiling:

- a) uning birinchi satrini;
- b) uning beshinchi satrini;
- c) uning birinchi 5 ta satrini;
- d) uning s_1 - nchisidan s_2 -gacha satrlarini;
- e) to'liq faylni.

538. 20 satrdan iborat matnli fayl mavjud. Berilgan tartib bo'yicha uning har bir satrini massivga yozing.

539. Matnli fayl mavjud. Unda "t" harfi bilan boshlanuvchi satr mavjudligini aniqlang. Agar bor bo'lsa, u holda bunday satrning tartib nomerini aniqlang.

540. Matnli fayl mavjud. Chop qiling:

- a) birinchi satrning birinchi elementini;

- b) birinchi satrning beshinchi elementini;
- c) birinchi satrning birinchi 10 ta belgilarini;
- d) birinchi satrning n -dan m -gacha belgilarini ($n < m$);
- e) ikkinchi satrning birinchi belgisini;
- f) n -nchi satrning k -nchi belgisini.

541. Matnli fayl mavjud. Uning satrlarini boshqa faylga ko‘chirib yozing. Ikkinchi faylning satrlari ketma-ketligi bo‘lishi kerak:

- a) berilgan fayl satrlari ketma-ketlik tartiblariga mos;
- b) berilgan fayl satrlari ketma-ketlik tartiblariga teskari tartibda.

542. Matnli fayl mavjud. Uning 30 ta belgidan ortiq satrlarini boshqa faylga ko‘chirib yozing.

543. Matnli fayl mavjud. Satrlardagi 0 belgisini 1 belgisi va teskarisiga almashtirib, ularni boshqa faylga ko‘chirib yozing.

544. Matnli fayl mavjud. Uning barcha juft satrlarini ikkinchi faylga, toqlarini esa uchinchi faylga yozing. Satrlarni kelish tartibi saqlansin.

545. Satrlari soni bir xil bo‘lgan ikkita matnli fayl berilgan. Satrlarni kelish tartibi saqlangan holda birinchi fayl satrlarini ikkinchisiga, ikkinchi fayl satrlarini birinchisiga ko‘chirib yozing. Yordamchi fayldan foydalanilsin.

546. Satrlari soni bir xil bo‘lgan ikkita matnli fayl berilgan. Ularning satrlari mos kelishini aniqlang. Agar mos kelmasa, u holda bu fayllardagi bir biridan farq qiluvchi birinchi satr nomerini aniqlang.

547. Bank mijozlari haqida ma’lumotlardan iborat matnli fayl yaratilgan. Mijozlar haqidagi ma’lumotlarni chop qiling.

548. Kichik rus harflari bilan yozilgan mantdan iborat fayl berilgan. Xuddi shu matnni bosh harflar bilan yozilganidan boshqa faylni hosil qiling.

MA'LUMOTLARNING MURAKKAB TOIFALARI

Ma'lumotlarning barcha toifalari oddiy bo'lsin, murakkab bo'lsin faqat bir toifadagi ma'lumotlarni saqlash uchun ishlatiladi. Masalan int toifasi faqat butun sonlarni saqlash uchun ishlatiladi. Massivlar va to'plamlarda esa elementlari qaysi toifada e'lon qilingan bo'lsa, faqat shu toifadagi ma'lumotlarni saqlaydi.

Amaliyotda esa axborotlarni saqlash, qayta ishlash uchun ma'lumotlarning har xil toifalarini aralashtirib ishlashga to'g'ri keladi. Ma'lumotlarning aralash toifasi bilan ishlash uchun C++ da struct (struktura) dan foydalanish mumkin.

Struktura deb, har hil toifadagi ma'lumotlarning cheklangan to'plamiga aytiladi. Strukturalar – maydon deb ataluvchi chekli sondagi hadlardan tashkil topadi.

Struktura nimaligini tushinish uchun quyidagi misolni ko'rib chiqamiz.

	Familiya Ism	Baholar
	Abdullaev Dilshod	5 5 4 5
	Abdurahimov Ne'mat	3 3 4 5
	Rejepova Dilbar	5 5 5 5
	Karimova Hafiza	4 4 5 5

Bu jadvalning har bir satri har hil toifadagi alohida elementlardan tuzilgan:

1. Tartib raqami – butun sonlardan
2. Familiya Ism – belgisi satrlardan
3. Baholar – butun sonlar massividan tashkil topgan.

Bu ma'lumotlarni bir guruhga, strukturaga birlashtirish mumkin.

Quyidagicha belgilashlar kiritamiz: Imtixon – strukturaning nomi; N – tartib raqami;

FI – Familiya Ism; B – baholar;

Strukturalarni e'lon qilish. Strukturalarni e'lon qilish uchun **struct** xizmatchi so'zidan foydalaniladi.

```
struct struktura_nomi {  
toifa_1 nom1;
```

```
toifa_2 nom2;  
...  
toifa_n nomn;  
};
```

Struktura elementiga murojaat:

Struktura orqali yangi ma'lumotlar toifasi hosil qilinadi. Dasturda strukturalardan foydalanish shu toifadagi o'zgaruvchilar e'lon qilinish orqali bo'ladi. Yuqoridagi misol uchun strukturani e'lon qilish quyidagicha bo'ladi:

```
struct imtixon {  
    int n;          // Tartib raqami  
    char FI[30];   // Familiya ismi  
    int b[4];      // Baholar  
};
```

Struktura elementlari (maydonlari) dasturda oddiy o'zgaruvchilar kabi ishlatiladi. Struktura maydoni elementlari ustida, uning toifasida nima amal bajarish mumkin bo'lsa shu amallarni bajarish mumkin. Strukturaning mayddoniga murojaat qilish uchun, o'zgaruvchidan keyin nuqta (.) qo'yiladi.

```
imtixon t; // talaba  
t.n = 1;  
t.b[0] = 5;  
t.b[1] = 5;  
t.b[2] = 4;  
t.b[3] = 5;  
strcpy(t.FI, "ErgashevBaxtiyor ");
```

Butun va haqiqiy toifadagi o'zgaruvchilarga qiymat berish oddiy amalga oshiriladi. Satrlarga qiymat berish esa, strcpy - funksiyasi orqali amalga oshiriladi.

Keling oddiy ishchi nomli struktura (sinf) hosil qilamiz va undan dasturda foydalanishni o'rganamiz. Ishchi strukturasi quyidagi maydonlarni o'z ichiga oladi:

- Familiy
- Ism
- Lavozim
- Oklad

- Yosh

Strukturaga ma'lumotlarni kiritish va chiqarishda amallarni qayta yuklashdan foydalanamiz. Amallarni qayta yuklashni, C++ da ob'yektga yo'naltirilgan dasturlash qismida batafsil to'xtalamiz. Xozircha asosiy e'tiborni strukturaga qaratamiz.

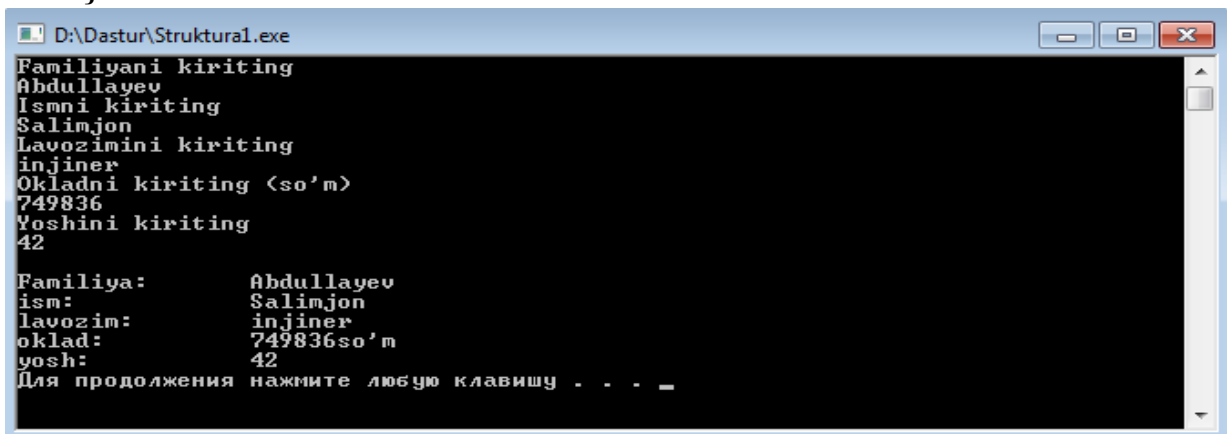
549. Ishchi strukturasin hosil qilish va dasturda foydalanish

```
#include <iostream>
using namespace std;
// ishchi sinfini e'lon qilish
struct ishchi{
char familiya[30];
char ism[30];
char lavozim[30];
float oklad;
int yosh;
};
// kiritish amalini qayta yuklash
istream& operator >> (istream& input, ishchi& k) {
cout << "Familiyani kiriting\n";
input >> k.familiya;
cout << "Ismni kiriting\n";
input >> k.ism;
cout << "Lavozimini kiriting\n";
input >> k.lavozim;
cout << "Okladni kiriting ($)\n";
input >> k.oklad;
cout << "Yoshini kiriting\n";
input >> k.yosh;
return input;
}
// chiqarish amalini qayta yuklash
ostream& operator << (ostream& out, ishchi k) {
out << "\nFamiliya:\t" << k.familiya;
out << "\nism:\t\t" << k.ism;
out << "\nlavozim:\t" << k.lavozim;
out << "\noklad:\t\t" << k.oklad << "$";
```

```

out << "\nyosh:\t\t" << k.yosh << endl;
return out;
}
int main() {
// ishchi sinfidagi ob'yektni e'lon qilish
ishchi p;
cin >> p;
cout << p;
return 0;
}

```



```

D:\Dastur\Struktural.exe
Familiyani kiriting
Abdullayev
Ismni kiriting
Salimjon
Lavozimini kiriting
injiner
Okladni kiriting (so'm)
749836
Yoshini kiriting
42

Familiya:      Abdullayev
ism:           Salimjon
lavozim:       injiner
oklad:         749836so'm
yosh:          42
Для продолжения нажмите любую клавишу . . .

```

550. n ta ishchi haqidagi ma'lumotlarni o'zida saqllovchi fayl hosil qiling. 30 yoshdan kichik bo'lgan ishchilarni chiqaruvchi dastur tuzilsin.

```

#include <cstdlib>
#include <iostream>
using namespace std;
// ishchi sinfini e'lon qilish
struct ishchi {
char familiya[30];
char ism[30];
char lavozim[30];
float oklad;
int yosh;
};
// kiritish amalini qayta yuklash
istream& operator >> (istream& input, ishchi& k) {
cout << "Familiyani kiriting\n";
input >> k.familiya;

```

```

cout << "Ismni kiriting\n";
input >> k.ism;
cout << "Lavozimini kiriting\n";
input >> k.lavozim;
cout << "Okladni kiriting (so'm)\n";
input >> k.oklad;
cout << "Yoshini kiriting\n";
input >> k.yosh;
return input;
}
// chiqarish amalini qayta yuklash
ostream& operator << (ostream& out, ishchi k) {
out << k.familiya;
out << " " << k.ism;
out << "\t" << k.lavozim;
out << "\t" << k.oklad << "so'm";
out << "\t" << k.yosh << endl;
return out;
}
int main() {
// ishchi sinfidagi ob'ektni e'lon qilish
ishchi p;
FILE * f;
int n = 0;
f = fopen("ishchi.dat", "a+");
cout<<"Faqat ro'yxatni ko'rish uchun 0 kiriting\n";
cout<<"Yangi kiritiladigan ishchilar sonini kiriting\n";
cin >> n;
if (n >= 1)
for (int i = 1; i <= n; i++) {
cout<<i<<" - ishchi ma'lumotlarini kiriting\n";
cin >> p;
// faylga ma'lumotlarni yozish
fwrite(&p, sizeof(ishchi), 1, f);
}
// fayl boshiga o'tish

```

```

rewind(f);
cout << "Ishchilar ro'yxati\n";
n = 0;
while (fread(&p, sizeof(ishchi), 1, f)) {
n++;
cout << n << " ";
cout << p;
}
// fayl boshiga o'tish
rewind(f);
cout<<"30 - yoshdan kichik ishchilar ro'yxati\n";
n = 0;
while (fread(&p, sizeof(ishchi), 1, f)) {
if (p.yosh <= 30) {
n++;
cout << n << " ";
cout << p;
}
}
fclose(f);
return 0;
}

```

```

D:\Dastur\Struct2.exe
Faqat ro'yxatni ko'rish uchun 0 kiriting
Yangi kiritiladigan ishchilar sonini kiriting
2
1 - ishchi ma'lumotlarini kiriting
Familiyani kiriting
To'rayev
Ismni kiriting
Bekzod
Lavozimini kiriting
Injiner
Okladni kiriting (so'm)
749836
Yoshini kiriting
35
2 - ishchi ma'lumotlarini kiriting
Familiyani kiriting
Alimov
Ismni kiriting
Zokir
Lavozimini kiriting
Hisobchi
Okladni kiriting (so'm)
642125
Yoshini kiriting
28
Ishchilar ro'yxati
1 To'rayev Bekzod Injiner 749836so'm 35
2 Alimov Zokir Hisobchi 642125so'm 28
30 - yoshdan kichik ishchilar ro'yxati
1 Alimov Zokir Hisobchi 642125so'm 28
Для продолжения нажмите любую клавишу . . . _

```

```

D:\Dastur\Struct2.exe
Faqat ro'yxatni ko'rish uchun 0 kiriting
Yangi kiritiladigan ishchilar sonini kiriting
0
Ishchilar ro'yxati
1 To'rayev Bekzod Injiner 749836so'm 35
2 Alimov Zokir Hisobchi 642125so'm 28
30 - yoshdan kichik ishchilar ro'yxati
1 Alimov Zokir Hisobchi 642125so'm 28
Для продолжения нажмите любую клавишу . . . _

```

Mustaqil ishlash uchun vazifa

551. Dastur yuklanayotganda ko'rsatilgan katalogda joylashgan C++ dastur nomlanishi va fayllar ro'yxatini ekranga chiqaruvchi dastur tuzing. Faraz qilamizki, birinchi satr dasturning maqsadi haqida ma'lumotlardan iborat izoh hisoblanadi.

552. Foydalanuvchi istagiga ko'ra dyuymdan santimetrga o'tkazilgan jadvalni ekranga, printerga yoki faylga chiqaruvchi dastur tuzing. Quyida dastur bajarilishi vaqtida taklif etiladigan ekran ko'rinishi keltirilgan (foydalanuvchi kiritgan ma'lumotlar yarim qalin shriftda berilgan).

*** Dyuymdan millimetrga o'tkazish jadvali ***

Natijani chiqarish:

- 1 – ekranga;
- 2 – printerga;
- 3 – faylga.

1 dan 3 gacha sonni kiriting va <Enter> ni bosing.

Sizning tanlovingiz: -> **1**

Dyuymlar	Millimetrlar
0.5	12.7
1.0	25.4
1.5	38.1
2.0	50.8
2.5	63.5
3.0	76.2
3.5	88.2
4.0	101.6
4.5	114.3

5.0

127.0

C++ DA GRAFIKA

C++ tilining grafik rejimida ishlash asoslari:

C++da grafik rejimida ishlash uchun maxsus graphics.h fayli mavjud. Bu direktiva o'zgarmlar, o'zgaruvchilar va turli qism dasturlardan tashkil topgan bo'lib, ular yordamida turli grafik adapterlar bilan xar xil tasvirlar chizish mumkin. Adapter kompyuterda graphics.h fayli bilan ishlash imkoniyatini yaratadigan maxsus qurilmadir. Grafik rejimiga o'tilganda ekran aloxida-aloxida nuqtalarga bo'linadi. Har bir nuqta o'z koordinatasiga egadir.

Eng ko'p ishlatiladigan adapterlar:

1. CGA - color graphics Adapter
2. MCGA - multi color graphics array
3. EGA - enhanced graphics Adapter
4. VGA - video graphics array.

Drayverlarni ko'rsatish uchun quyidagi o'zgarmlar ishlatiladi:

Detect = 0 CGA = 1; MCGA = 2; EGA=3; VGA=9.

Matn rejimidan grafik rejimiga utish uchun maxsus proseduradan foydalaniladi:

initgraph (&gd, &gm, "path");

Bu yerda:

gd - drayver nomi;

gm - rejim nomi;

Path - kerakli drayver faylining yo'li.

initwindow(x, y);

Bu yerda:

x va y hosil qilinadigan oynaning o'lchami

Ko'pincha gd=0 deb olinadi. Drayverlar .bgi fayllarida saqlanadi. Agar drayver ishchi katalogning o'zida joylashgan bo'lsa, u xolda Path = " " (bush belgisi) bo'ladi. Grafik rejimidan yana matn rejimiga utish kerak bo'lsa, closegraph() funksiyasi ishlatiladi.

Chizmalarni hosil qilish uchun ishlatiladigan prosedura va funksiyalar:

1. putpixel(x, y, color) - x va y koordinatadagi nuqtani color rangda chizish;

2. `getpixel(x, y)` - x va y koordinatadagi nuqtaning rangini aniqlaydi;

3. `line(x1, y1, x2, y2)` - $x1$ va $y1$ koordinatadagi nuqtadan $x2$ va $y2$ koordinatadagi nuqttagacha kesma chizish;

4. `circle(x, y, r)` - markazi x va y koordinatada va radiusi R bo'lgan aylana chizish;

5. `rectangle(x1, y1, x2, y2)` - yuqori chap nuqtasi $x1$ va $y1$ koordinatada, o'ng pastki nuqtasi $x2$ va $y2$ koordinatada bo'lgan tug'rito'rtburchakni chizish;

6. `setbkcolor(color)` - orqa fonga rang berish;

7. `setcolor (color)` - chizish rangini o'rnatish (rangli qalam); Bu yerda `color` - rang nomeri yoki nomi. Agar rang nomi yoziladigan bo'lsa, uni katta xarflarda yoziladi.

8. `bar(x1, y1, x2, y2)` - joriy rang va chiziqlar yordamida ichi bo'yalgan tug'rito'rtburchak chizish;

9. `fillellipse(x, y, xr, yr)` - markazi x va y da, xr kenglikda va xr balandlikda ichi bo'yalgan rangli ellips chizadi;

10. `getmaxx` - joriy rejim va drayverlar uchun nuqtalar sonini aniqlash;

`getmaxy` - joriy rejim va drayverlar uchun vertikal nuqtalar soni. Bu prosedura yordamida kompyuterning o'zi ekrandagi maksimal nuqtalar sonini aniqlaydi.

11. `linerel(x, y)` - x va y koordinatali nuqtadan joriy nuqttagacha kesma chizish;

`lineto(x, y)` - joriy nuqtadan x va y koordinatali nuqttagacha kesma chizish;

12. `bar3D(x1, y1, x2, y2, h, top)` - parallelopiped chizadi. Bu yerda h - parallelopipedning uzunligi; `top` - yuqori qismini chizish uchun kerak. Agar `topon` - bo'lsa tomi bor, agar `topoff` - bo'lsa tomi yo'q.

13. `arc(x, y, a, b, r)` - yoy chizish uchun. Bu yerda x va y - markazning koordinatalari, a - bosh burchak, b - oxirigi burchak, r - yoy radiusi. Burchaklar gradusda qabul qilinadi.

14. `ellipse(x, y, a, b, xr, yr)` - xuddi shu tartibda ellips yoyini chizadi.

15. drawpoly(n, p) – ko‘pburchak chizish uchun. Bu yerda n – ko‘pburchakning uchlari soni; p -. ko‘pburchak uchlarining koordinatalari.

Bo‘yash va chizish usullari hamda stillari haqida:

1. setfillstyle(style, color) - bo‘yash usul va rangni o‘rnatish. Bu yerda style - o‘zgarmas kattalik bulib, u quyidagicha bulishi mumkin:

- 0 - soxani fon rangi bilan to‘ldirish;
- 1- soxani rang bilan uzluksiz to‘ldirish;
- 2 - qalin gorizontal chiziqlar
- 3 - ingichka og‘ma chiziqlar
- 4 – yug‘on og‘ma chiziqlar
- 5 – yug‘on og‘ma chiziqlar (boshqa stil)
- 6 – og‘ma yo‘llar
- 7 – to‘rtburchakli chiziqlar
- 8 – og‘ma to‘rtburchaklar
- 9 - zich og‘ma shrtixlar
- 10 - siyrak nuqtalar (u yer - bu yerda)
- 11 - zich nuqtalar bilan

2. floodfill(x, y, color) - joriy rang va usuldan foydalangan xolda chegaralangan soxani bo‘yash. Bu yerda x va y - shu soxaga tegishli bo‘lgan biror nuqta koordinatasi. Avval rang, keyin stili ko‘rsatiladi.

Masalan:

```
setcolor(4); {qizil rangli qalam, chegara rangi}  
setfillstyle(1,2);{1-stil bilan yashil rang bilan bo‘yash}  
circle(50,50,35); {radusi 35 bo‘lgan aylana chizish}  
floodfill(50,50,4); {aylana ichiga rang tukish, bo‘yaladigan  
chegara rangi rangli qalam bilan bir xil bulishi kerak}
```

3. setlinestyle(s,a,b) – turli stildagi chiziqlarni chizish uchun;

Bu yerda s - style nomeri; a – foydalanuvchi stilini yaratishi mumkin bo‘lgan parametr, odatda a=1 deb olinadi; b- chiziqning qalinligini ko‘rsatadigan parametr

- 0 – oddiy chiziq;
- 1 – mayda punktir chiziq;
- 2 – qalin va uzunchok punktir chiziq

3 – yupka va uzunchok punktir chiziq;

4 – siyrak nuqtali chiziq.

Grafik rejimida shriftlar. Grafik rejimi xolatida turli shriftlardan foydalanib matnlarni xam yozsa bo‘ladi. Shriftlar .chr kengaytmali fayllarda saqlanadi. Ular .bgi fayllari bilan bitta katalogda saqlanishi shart.

1. outtextxy(x, y, 'matn'); - matnni yozish; bu yerda x va y matn boshlanadigan nuqta koordinatalari; masalan: outtextxy(10, 10, 'Mirzaev K. 212-07 Aty');;

2. settextstyle(sh, n, r); - matn shriftini o‘rnatish; bu yerda sh - shrift nomeri (0 - vektorli shrift, 1 - standart shrift); n - shrift yunalishi (0 - chapdan o‘ngga, 1 - quyidan yuqoriga yozish); r - shrift o‘lchami (oddiy shriftda 1, vektorli shriftda 4 deb olinadi);

3. settextjustify(h, v); - yozilgan qatorni tekislaydi. U outtextxy prosedurasidan keyin yoziladi. Bu yerda h - gorizontal tekislash; v - vertikal tekislash; Gorizontal tekislash uchun: 0 - chapga; 1 - markazga; 2-o‘ngga. Vertikal tekislash uchun: 0 - pastga; 1 - markazga; 2 - yuqoriga.

4. setUserCharSize - vektor shriftlari uchun bir xil belgilarning eni va buyini o‘rnatadi. Masalan: setUserCharSize(x1, y1, x2, y2);

Ekranda chizmalarni xarakatlantirish. Ekranda chizmalarni xarakatlantirish uchun kerak bo‘ladigan proseduralar:

1. imagesize(x1, y1, x2, y2) – ekranning chap yuqori nuqtasi va o‘ng pastki nuqtasi koordinatalaridan tug‘rito‘rtburchakli soxani saqlash uchun kerak bo‘ladigan xotiraning ulchami (baytlarda olinadi);

2. getimage(x1, y1, x2, y2, p) – dinamik xotiraning berilgan r maydonida tug‘rito‘rtburchakli tasvirni saqlash. Bu yerda r – tasvir saqlanadigan joyning adresini saqlaydigan o‘zgaruvchi, ya’ni ko‘rsatkich.

3. putimage(x, y, p, m) – ekranning berilgan joyiga tasvirni chiqarish; bu yerda x va y – xotiraning p maydonidagi tasvirdan nusxa kuchiriladigan ekran maydonining chap yuqori nuqtasi; m – tasvirni ekranga chiqarish rejimi. Agar:

m = 0 (NormalPut) - tasvirni kuchirish. Bunda eskisi uchib, yangisi paydo bo‘ladi (xuddi yurib ketayotgandek)

m = 1 (XorPut)
m = 2 (Orput) –
m = 3 (AndPut)

553. Ekran bo‘ylab sariq aylanachalar, parallelepiped va ellips tasvirlarini hosil qilish.

```
#include <cstdlib>
#include <iostream>
#include <graphics.h>
#include <stdlib.h>
using namespace std;
int main() {
    initwindow(600, 500);
    setcolor (14); // sariq qalam
    for(int i=0; i<=20; i++)
        for (int j=0; j<=20; j++)
            circle(i*50,j*30,55); //sariq rangli aylanalar
    rectangle(0,0,getmaxx(),getmaxy()); //ekran bo‘ylab //to‘g‘ri
    to‘rtburchak
    setcolor(11); // to‘q feruza rangli qalam
    bar3d(200,300,100,150,30,1);
    //parallelopiped, ichi oq
    setcolor(CYAN); // och feruza rangli qalam
    fillellipse(350, 360, 135, 90);
    //ellips, ichi oq rangda
    closegraph();
    return 0;
}
```

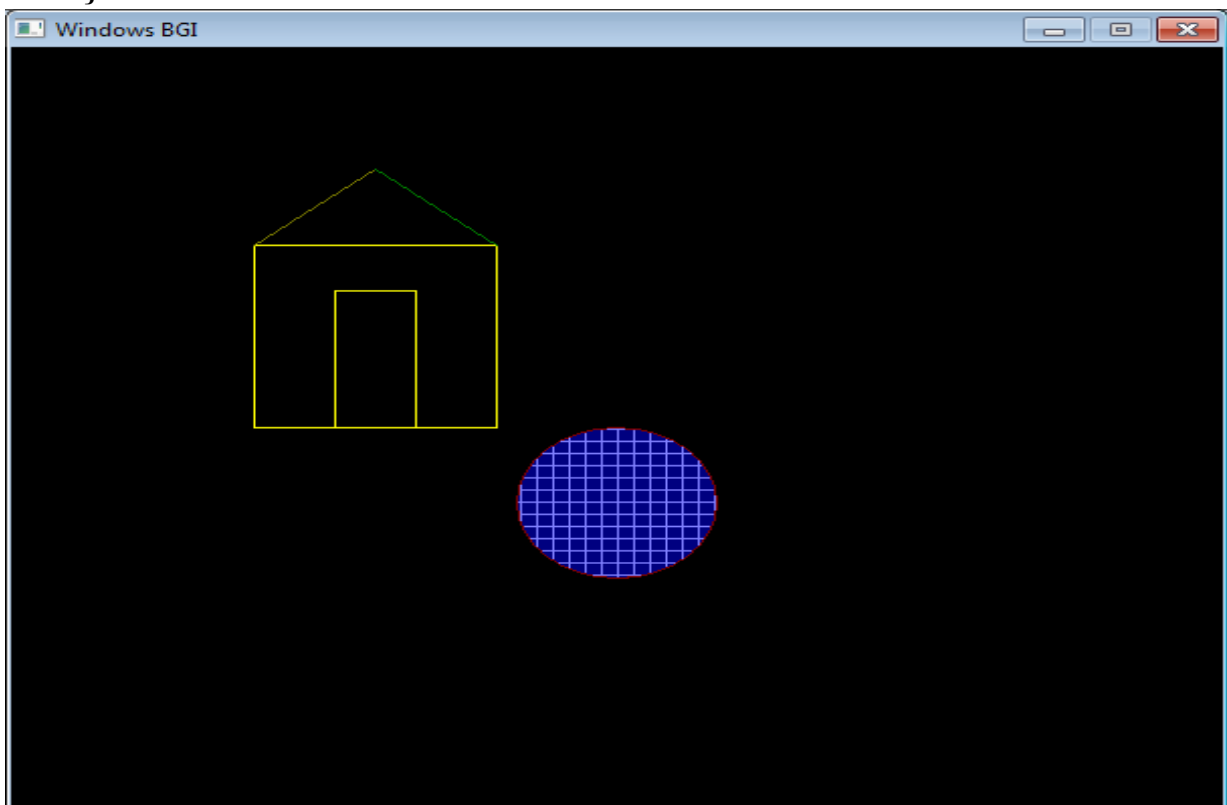
554. Uy tasvirini va ko‘k katakchali aylanani hosil qilish.

```
#include <cstdlib>
#include <iostream>
#include <graphics.h>
#include <stdlib.h>
using namespace std;
int main() {
    initwindow(600, 500);
    setbkcolor (BLUE);
```

```

setcolor (14);
rectangle (120, 130, 240, 250);
setcolor (6);
line (120, 130, 180, 80);
setcolor (2);
line (180, 80, 240, 130);
setcolor (14);
rectangle (160, 160, 200, 250);
setcolor (4);
setfillstyle(7, 9);
circle( 300, 300, 50);
floodfill (300, 300, 4);
closegraph();
return 0;
}

```



555. Funksiyalarning grafiklarini chizish.

```

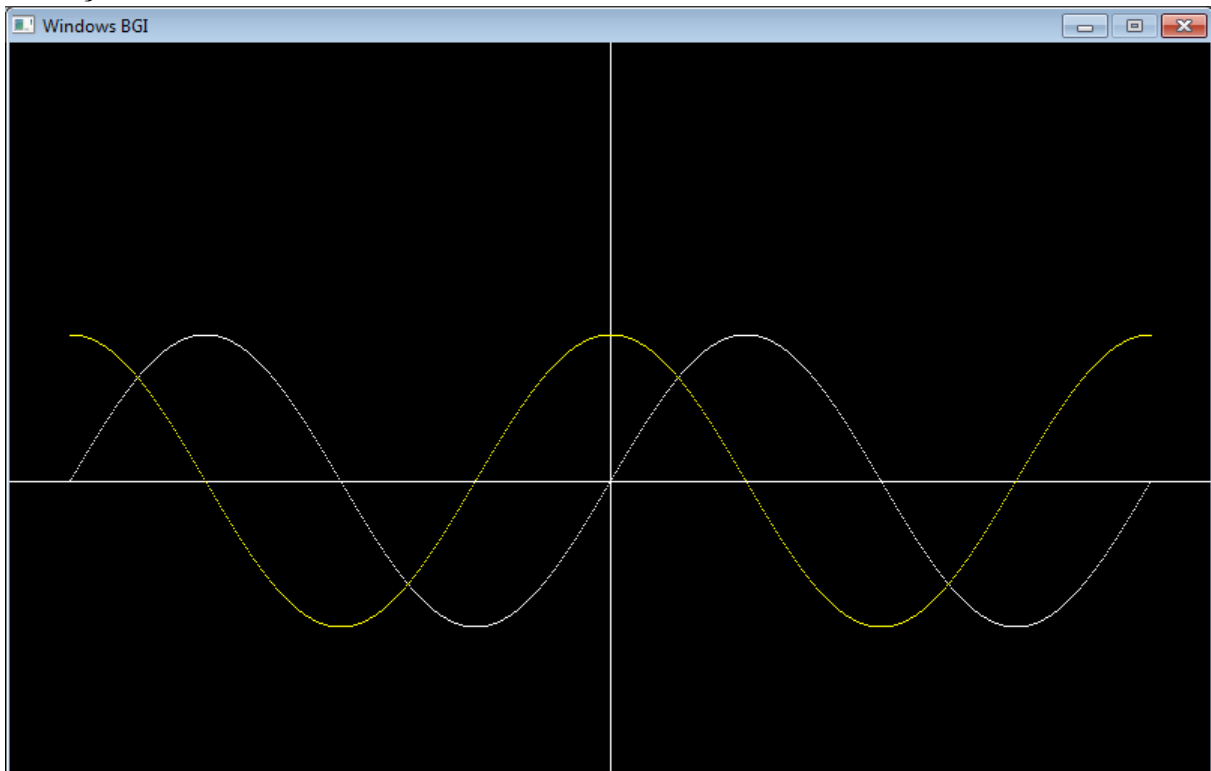
#include <graphics.h>
#include <math.h>
int main() {
initwindow(800, 500);
line(0, 300, getmaxx(), 300);

```

```

line(400, 0, 400, getmaxy());
int x, y;
float pi = 3.1415;
for (int a = -360; a <= 360; a++) {
x = 400 + a;
y = 300 - sin(a * pi / 180) * 100;
putpixel(x, y, WHITE);
y = 300 - cos(a * pi / 180) * 100;
putpixel(x, y, YELLOW);
delay(10);
}
closegraph();
return 0;
}

```



556. Sekunt va minut ko'rsatkichi bor bo'lgan harakatlanadigan soat tasvirini ekranga chiqaradigan dastur tuzing.

```

#include <cstdlib>
#include <iostream.h>
#include <graphics.h>
using namespace std;
int main() {
float soat,min,d,s;

```

```

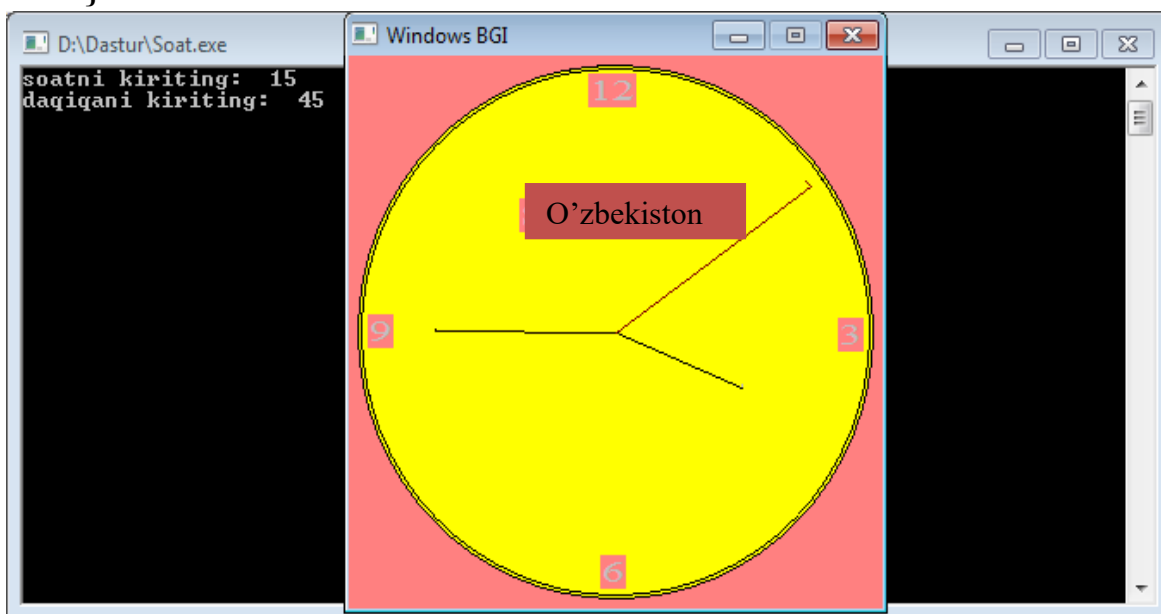
cout<<"soatni kiriting: ";
cin>>soat;
cout<<"daqiqani kiriting: ";
cin>>min;
while (soat>=12)
soat=soat-12;
while (min>=90)
min=min-90;
d=360-(6*min-90);
s=360-(soat*30-90+0.008333333333333*min*60);
initwindow(310,310);
struct arccoordstype q;
int g=90;
setbkcolor(LIGHTRED);
while (g>=0) {
cleardevice ();
setcolor(BLACK);
setfillstyle (1,YELLOW);
circle(155,155,150);
floodfill(155,155,BLACK);
circle(155,155,148);
setcolor(7);
settextstyle(6,0,1.9);
outtextxy(100,80,"O‘zbekiston");
outtextxy(140,10,"12");
outtextxy(285,147,"3");
outtextxy(147,280,"6");
outtextxy(12,145,"9");
arc(155, 155, s, s+1, 80);
getarccoords(&q);
setcolor(BLACK);
line(155, 155, q.xstart, q.ystart);
s-=0.008333333333333;
if (s<=0)
s=360;
setcolor (16);

```

```

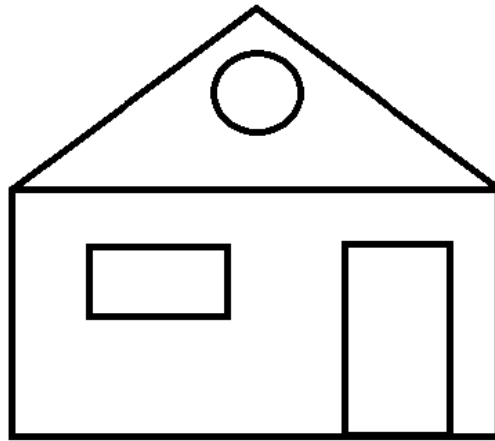
arc(155,155,d,d+1,105);
getarccoords (&q);
line (155, 155, q.xstart, q.ystart);
d-=0.1;
if (d<=0)
d=360;
setcolor (RED);
arc(155,155,g,g+1,140);
getarccoords(&q);
line(155, 155, q.xstart, q.ystart);
g-=6;
if (g<=0)
g=360;
delay(1000);
}
closegraph ();
return 0;
}

```

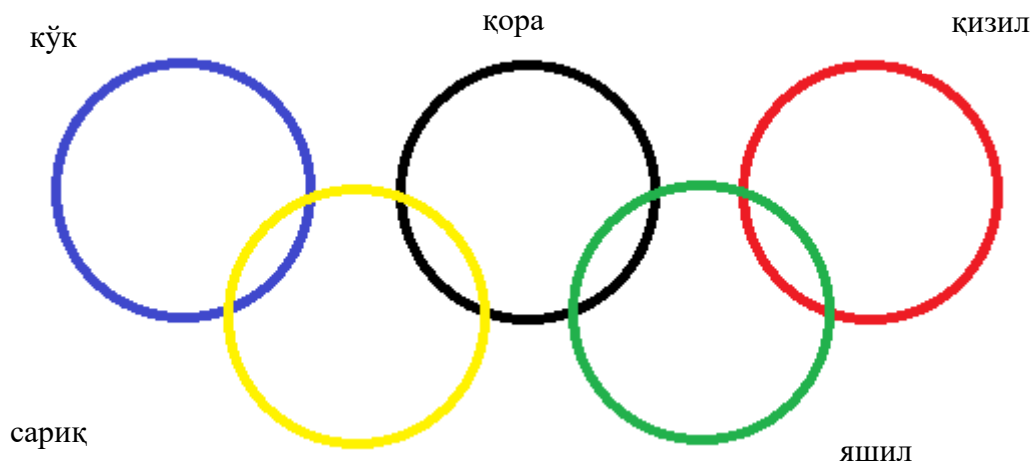


Mustaqil ishlash uchun vazifa

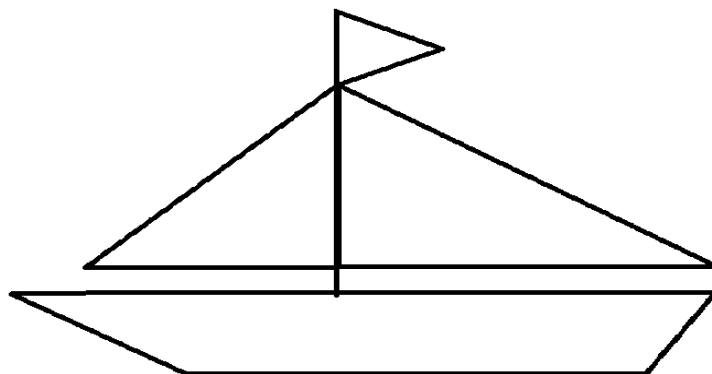
557. Ekranda uychni tasvirini hosil qiladigan dastur tuzing.



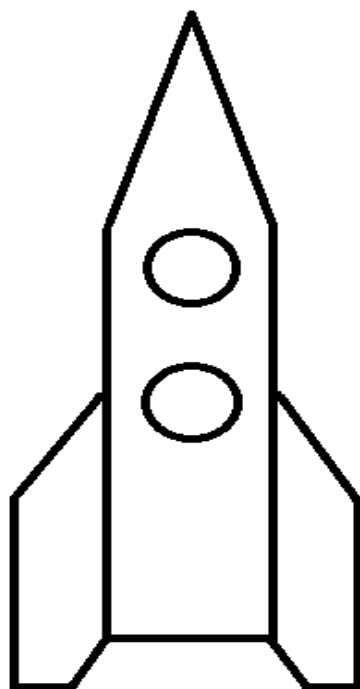
558. Ekranga Olimpiada o'yinlari bayrog'ini chiqaradigan dastur tuzing. Bayroq tasviri quyida keltirilgan (bitta katak besh pikselga mos keladi).



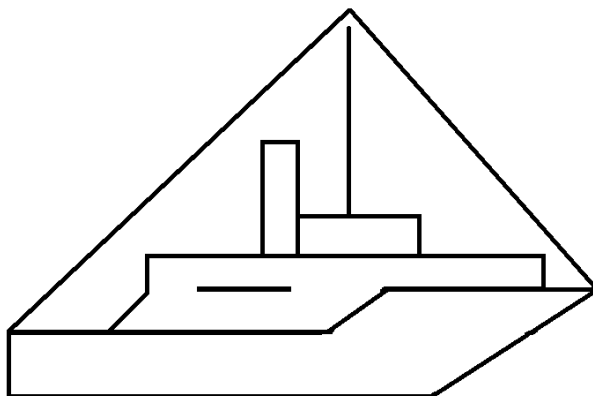
559. Ekranda kemani tasvirini hosil qiladigan dastur tuzing.



560. Ekranda raketani tasvirini hosil qiladigan dastur tuzing.



561. Ekranga kema tasvirini bazaviy (boshlang'ich) nuqta usulidan foydalanib chiqaradigan dastur tuzing.

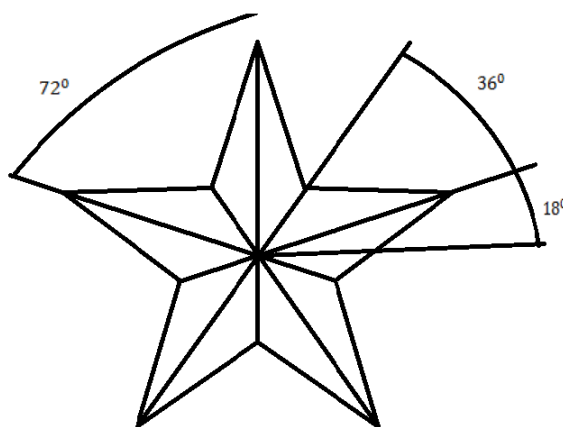


562. Tasodifiy diametrli va rangli 100 ta aylanadan iborat bezakni ekranda hosil qiladigan dastur tuzing.

563. Tasodifiy o'lchamdagi va rangdagi 50 ta to'rtburchakdan iborat bezakni ekranda hosil qiladigan dastur tuzing.

564. Tasodifiy rangga ega bo'ladigan va tasodifiy koordinatalarga ega bo'lgan, 100 ta bo'g'inlardan iborat bo'lgan siniq chizik chizadigan dastur tuzing.

565. Besh yulduz konturini ekranga chiqaruvchi dastur tuzing.



566. Oq chiziq bilan okontovkali qizil rangdagi besh yulduz konturini ekranga chiqaruvchi dastur tuzing.

567. Ekranda oltiburchakni hosil qiladigan dastur tuzing.

568. O‘zbekiston Respublikasining davlat bayrog‘ini ekranga chizadigan dastur tuzing.

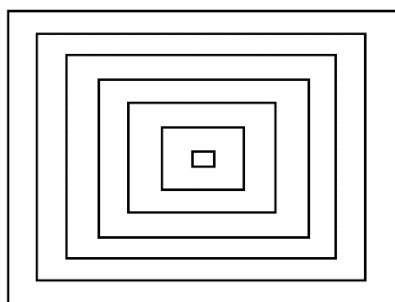
569. Sariq rangdagi sho‘x siymoni ekranda tasvirlovchi dastur tuzing.



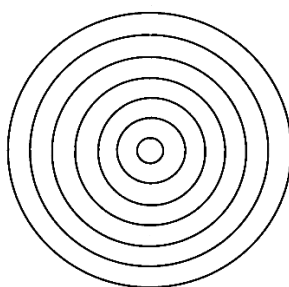
570. Sariq rangdagi qayg‘uli siymoni ekranda tasvirlovchi dastur tuzing.



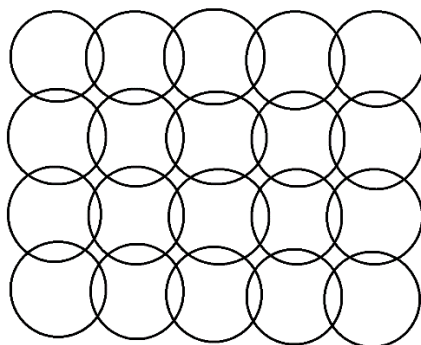
571. Quyidagi naqsh tasvirini ekranga chiqaradigan dastur tuzing.



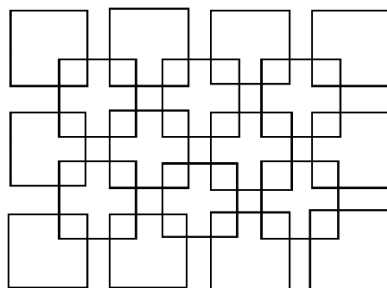
572. Quyidagi naqsh tasvirini ekranga chiqaradigan dastur tuzing. Aylanalar har xil rangda bo'lishi kerak: ko'k rangdan oq ranggacha (ma'lumotnomadan setcolor funksiyasining ranglarning kodlar jadvalini qarang).



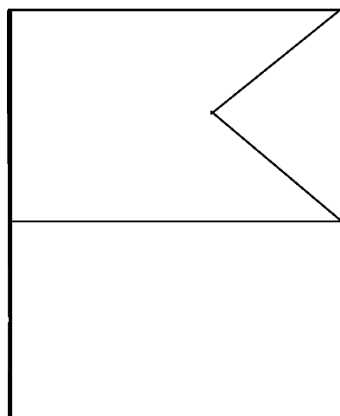
573. Quyidagi naqsh tasvirini ekranga chiqaradigan dastur tuzing.



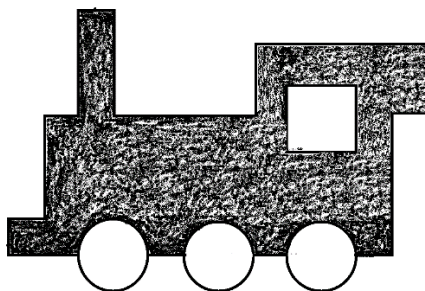
574. Quyidagi naqsh tasvirini ekranga chiqaradigan dastur tuzing.



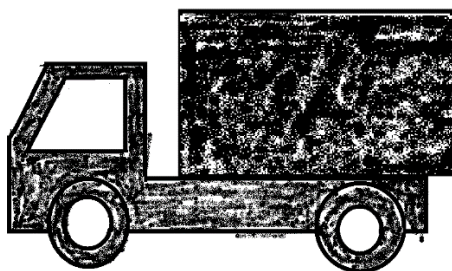
575. Ekranda qizil rangdagi bayroq tasvirini chizadigan dastur tuzing.



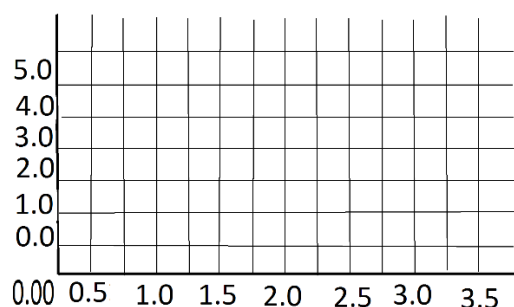
576. Ekranga parovoz rasmini chizadigan dastur tuzing. Asosiy nuqta usulidan foydalaning.



577. Ekranda avtomobilni chizadigan dastur tuzing. Avtomobilning g'ildiraklarini hosil qilishni ta'minlaydigan ko'rsatmalarni funksiya kabi rasmiylashtiring.



578. Raqamlangan koordinata to'rini ekranga chiqaradigan dastur tuzing.



579. $y=0,5x^2+4x-3$ funksiyaning nuqtali grafigini ekranga chiqaradigan dastur tuzing. Argument -15 dan 5 oraliqda o‘zgaradi; argument qadami – 0,1. Grafik, kesishish nuqtasi ekran markazida bo‘ladigan koordinata o‘qlari kesishishida tasvirlanishi kerak.

580. Ekranda aylananing harakatini chizadigan dastur tuzing.

581. Ekranda kemani chizadigan funksiyaning tuzing. Funksiya parametri sifatida keyingi chiziladigan kemandagi bazaviy nuqta koordinatalari va rangi bo‘lishi kerak. Bu funksiya foydalanib ekranda kemandagi harakatini hosil qiladigan dastur tuzing.

582. Ekranda sinfning o‘zlashtirish gistogrammasini chiqaradigan dastur tuzing, masalan, nazorat ishi yakuni bo‘yicha. Boshlang‘ich ma‘lumotlar ekranning matnli tartibida kiritilishi kerak. Dastur ishlash vaqtida taklif etilgan ekran ko‘rinishi quyida keltirilgan. Quyida boshlang‘ich ma‘lumotlarni kiritish vaqtidagi va undan keyin diagramma ko‘rinishidagi ekran ko‘rinishi keltirilgan.

Nazorat ish natijalarini qayta ishlash.

Boshlang‘ich ma‘lumotlarni kiritish:

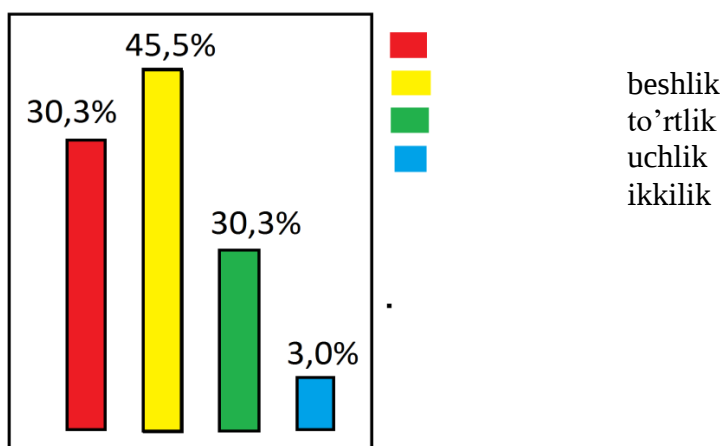
Beshlik ->10

To‘rtlik ->15

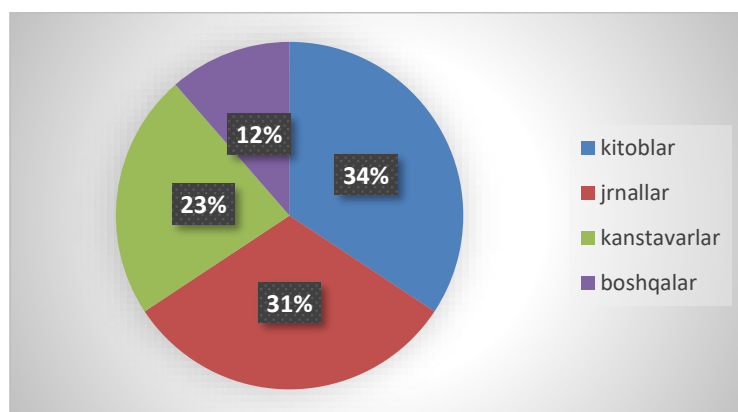
Uchlik ->7

Ikkilik->1

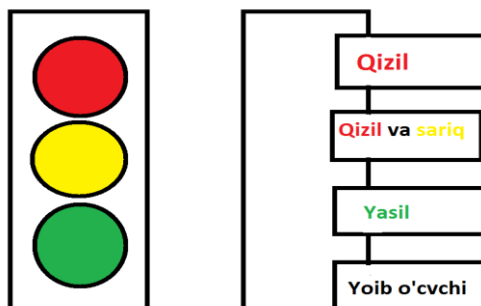
Nazorat ish natijalari



583. Kitob magazinidagi tovar aylanishini (foizlarda) tasvirlaydigan aylana diagrammani chiqaradigan dastur tuzing. Boshlang'ich ma'lumotlar (kategoriya bo'yicha sotilganlar xajmi so'mda: kitoblar, jurnallar, otkrytkalar va kanstovarlar) dastur ishlash vaqtida kiritiladi. Diagrammaga misol quyida keltrilgan.



584. Ishlayotgan sfetofor tasvirini ekranga chiqaradigan dastur tuzing. Taklif etilgan sfetofor ko'rinishi va uning ishlash algoritmi quyida keltirilgan.



585. $y=2\sin(x)e^{x/5}$ funksiyaning grafigini ekranga chiqaradigan dastur tuzing.

FOYDALANILGAN ADABIYOTLAR

1. SH. Fozilov, S.Jumanazarov, D.Muhamediyeva. Dasturlash asoslari.-Toshkent 2006. – 200 b
2. A.B.Qarshiyev, O.R.Yalgashov. Paskal tilida dasturlash bo'yicha amaliy va laboratoriya mashg'ulotlari uchun uslubiy qo'llanma.- Samarqand 2006. – 100 b
3. A.B.Qarshiyev, Z.M. Maxmudov. «C++da dasturlash» bo'yicha laboratoriya mashg'ulotlari uchun uslubiy ko'rsatmalar. – Samarqand 2006. – 36 b
4. R.A.Yusupov, SH.Mirzanova, S.N.Iskandarova. C/C++ tilida masalalar va topshiriqlar. – Samarqand 2013.- 223 b
5. R.A.Yusupov, A.A. Ismoilov, S.N.Iskandarova. Obyektga yo'naltirilgan dasturlash tillari fanidan ma'ruzalar matni. Samarqand 2013. – 90 b
6. A.B.Qarshiyev, G.T.Safarova. Delphi muxitida dasturlash bo'yicha laboratoriya ishlarini bajarish uchun uslubiy kursatmalar(1-qism). Samarqand 2007. – 50 b
7. A.A. Ismoilov, A.Qayumov, M.Karimov. dasturlash asoslari fanidan laboratoriya mashg'ulotlarini bajarish uchun uslubiy ko'rsatma. Samarqand 2014. – 52 b
8. C ++ o'bekтли dasturlash tili bo'yicha multemediali elektron o'quv qo'llanma
9. Murray Hill. The C++ Programming language. Bjarne Strustrup AT&T labs

MUNDARIJA

Kirish.....	3
Shaxsiy kompyuter arxitekturası. Xavfsızlık texnikası, kompyuter sinfida ishlash qoidalari.....	4
Axborotni ifodalash usullari. Axborot o‘lchov birliklari..	11
Kompyuterlarning arifmetik asoslari. Sanoq sistemalari (2lik, 8lik, 16lik), ularda amallarni bajarish.....	19
Algoritmlar, xossalari, ularni ifodalash usullari.....	25
Dasturlashning birinchi qadamlari.....	30
Dastur va dasturchilar haqida.....	30
C++ dasturlash tilining tarixi.....	32
C++ tilinig tashkil etuvchilari.....	32
O‘zgaruvchilar (variables).....	34
C++ da arifmetik amallar.....	35
C++ da dastur tuzilishi.....	40
Chiziqli jarayonlarni dasturlash.....	44
Tarmoqlanuvchi jarayonlarni dasturlash.....	80
Shartli operatorlar.....	80
Tanlash operatori.....	81
Takrorlanuvchi jarayonlar bilan ishlash.....	124
Sikl operatori.....	124
Algebraik va transcendent tenglamalarni yechish.....	142
Chekli yig‘indini hisoblash.....	152
Oddiy massivlar bilan ishlash.....	159
Aniq integrallarni hisoblash.....	180
Differensial tenglamalarni yechish.....	189
Ko‘p o‘lchovli massivlar bilan ishlash.....	196
Massivlar ustida amallar bajarish.....	206
Funksiyalar.....	229
Satrlar va satrlar ustida amallar bajarish.....	237
Fayllar bilan ishlash.....	250
Matnli fayllar.....	259
Ma’lumotlarning murakkab toifalari.....	267
C++ da grafika.....	275
Foydalanilgan adabiyotlar ro‘yxati.....	291

**Ismoilov A.A., Yusupov R.A., Boymurodov.,
A.X. Xurramov A.J.**

C++ da dasturlash asoslari

O'quv qo'llanma

Toshkent – “Ishonchli hamkor” – 2021

Muharrir:	<i>O.Rahimov</i>
Tex.muxarrir:	<i>N.Ismoilova</i>
Musahhih:	<i>M.Yusupova</i>
Kompyuterda sahifalovchi:	<i>D.Usmanova</i>

Nashriyot litsenziya № 2044, 25.08.2020 y.

Bosishga ruxsat etildi 30.07.2021 y.

Bichimi 60x84 ¹/₁₆. “Times new roman” garniturası. Offset
bosma usulida bosildi. Shartli bosma tabog‘i 18,5.

Adadi 100 dona.

Buyurtma № 307.

Zebo prints MCHJda chop etildi.

Manzil: Toshkent shahar, Yashnobod tumani,
22-harbiy shaharcha

[illegible]

[illegible]