

**O'ZBEKISTON ALOQA, AXBOROTLASHTIRISH VA
TELEKOMMUNIKATSIYA TEXNOLOGIYALARI DAVLAT QO'MITASI**

TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI

KOMPYUTER INJINIRING FAKULTETI

“Informatika asoslari” kafedrası

**“INFORMATIKA” FANIDAN LABORATORIYA ISHLARINI
BAJARISH UCHUN USLUBIY KO'RSATMALAR**

Maxsus fakultet talabalari uchun

Toshkent – 2015

Mualliflar: dotsent Q.S.Raxmanov, katta o'qituvchi Sh.T.Qosimova, assistent Sh.B.Abidova. "Informatika" fanidan laboratoriya ishlarini bajarish uchun uslubiy ko'rsatmalar (Maxsus fakultet talabalari uchun). TATU, Toshkent 2015y. 156 b.

Taqdim etilayotgan ushbu uslubiy ko'rsatma o'z ichiga 9 ta laboratoriya ishini qamrab olgan.

2013-2014 o'quv yilida maxsus fakultet tomonidan o'quv dasturi o'zgartirilish taklifi tushdi va mos ravishda "Informatika" fani bo'yicha uslubiy ko'rsatma ishlab chiqish zarurati tug'ildi. Maxsus fakultet 1-kurs talabalari birinchi va ikkinchi semestrlarda "Informatika" fanini o'zlashtirishda quyidagi mavzularni o'rganadilar:

- Kompyuterning asosiy qurilmalari;
- Sanoq sistemalari haqida tushuncha;
- C++ tilining asosiy konstruksiyalari;
- C++ tilining operatorlari;
- Massivlar va ular bilan ishlash;
- C++ tilining grafik imkoniyatlari;
- Funksiyalarni tashkil etish;
- Borland C++ Builder 6 kutubxonasi komponentalari.

Uslubiy ko'rsatmada har bir laboratoriya ishi mavzusi bo'yicha ma'lumot, ko'rsatmalar, misollarning bajarilishi va variantlar bilan ta'minlangan.

Taqrizchi

O'zMU, "Informatika va tadbiiy dasturlash" kafedrasida dotsenti, f.-m.f.n.

A.Xaydarov

Uslubiy ko'rsatma 2015 yilning 8-yanvaridagi 18-sonli kafedra majlisida muhokama qilindi va bosmaga chiqarishga tavsiya etildi.

"Informatika" fani bo'yicha laboratoriya ishlarini bajarish uchun uslubiy ko'rsatma TATU ilmiy-uslubiy kengashi "___" _____-dagi ___-sonli majlisida ko'rib chiqildi va nashr qilishga ruxsat berildi.

© Toshkent axborot texnologiyalari universiteti 2015 yil

1-Laboratoriya ishi. Shaxsiy kompyuter arxitekturasi

Ishning maqsadi

1. “Orion” o’qituvchi dasturning qo’llanishi.
2. Kompyuterning asosiy qurilmalari. Hotira qurilmasi. Ularda axborotni saqlash usullari.
3. Kompyuterning dasturiy ta’minoti.
4. Kompyuterning tashqi qurilmalari, ko’rinishi, qo’llanishi.

Topshiriq

1. Har bir talaba ”Orion” o’qituvchi dasturini o’rganib, olgan bilimlarini tatbiq eta olishi lozim.
2. Hisobot shaklida har bir bo’lim bo’yicha qisqacha konspekt keltirilishi kerak.
3. 1-laboratoriya bo’yicha beriladigan barcha vazifalarni bitta hisobot shaklida topshiriladi.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. “Orion” dasturidan olgan va zarur adabiyotlardan to’ldirilgan matn.

Nazariy qism

Kompyuter (Computer) inglizchadan o’zbekchaga o’girilganda “hisoblovchi” ma’nosini bildiradi, ya’ni hisoblash ishlarini bajaruvchi qurilma.

Inson ongli faoliyati davomida o’zining jismoniy ishlarini bajaradigan har xil texnik qurilmalar, asbob- uskunalar yaratishga harakat qilgan. Shu jumladan, aqliy faoliyat ishlarini ham bajaradigan, asosan hisob - kitob ishlarini bajaradigan texnikaviy qurilmalarni yaratishga harakat qilgan. Eng qadimgi hisob-kitob asboblardan biri schetdir.

1642 yilda fransuz olimi Blez Paskal qo’shish amalini bajaradigan mexanik

qurilma ixtiro qildi.

1642 yili fransuz matematik va fizik olimi B. Paskal qo'shish va ayirish amallarini bajaradigan mexanik mashina yaratgan.

1673 yilda olmon olimi Vilgelm Leybnis to'rt arifmetik amallarni bajaradigan mexanik arifmometrni yaratdi.

XIX asrda arifmometr juda keng tarqaldi. Uni hatto eng murakkab hisob ishlariga ham qo'llay boshlashdi. Bunday hollarda oldin qilinadigan ishlarining bajarilish ketma - ketlik qo'llanmasi to'liq yozilib olinar va shu asosda ish bajarilar edi. Odatda bu qo'llanmani bajariladigan ishning dasturi deyiladi.

1834 yili angliyalik olim Ch. Bebbidj analitik mashina ixtiro qildi. Bu mashina dastur asosida ishlaydigan birinchi hisoblash mashinasining loyihasi edi.

Keyinchalik 1883 yili Ch.Bebbidj hozirgi zamondagi hisoblash mashinasining g'oyasini va loyihasini yaratdi, ammo o'sha davr texnika darajasi bunday mashinalarni yaratish imkonini bermas edi. Uning fikriga ko'ra bu qurilma hamma hisoblash ishlarini insonning ishtirokisiz o'zi avtomatik ravishda bajarishi kerak edi. Buning uchun u qurilma hisoblash ishining dasturini tushunib, shu dastur asosida hamma ishlarni bajara olishi kerak edi.

Sonli hisoblash mashinasi tuzilishining asosiy prinsiplarini amerikalik matematik Djon fon Neyman, G.Goldsteyn va A.Berks ishlab chiqdilar. Ularning va Ch.Bebbidj g'oyasi bo'yicha hisoblash mashinalarining ishlashi ikki prinsipga asoslanishi kerak:

- Masalani insonning ishtirokisiz, dastur asosida olib borilishi;
- Masalani yechish uchun kerak bo'ladigan hamma boshlang'ich va oraliq ma'lumotlar hamda masalani yechish dasturlarini saqlab turishi.

Buning uchun yaratilajak hisoblash mashinasi quyidagi qurilmalardan iborat bo'lishi lozim edi:

- Boshlang'ich ma'lumotlarni, oraliq qiymatlarni hamda masalani yechish dasturini saqlab turadigan qurilma. Hozirda bunday qurilmani xotira deb yuritiladi;
- Ish bajaradigan qurilma. Odatda uni arifmetik - mantiqiy qurilma deyiladi;
- Dastur bo'yicha ish bajaradigan va qurilmalarning ishlashini insonning

ishtirokisiz boshqarib boradigan qurilma. Uni boshqarish qurilmasi deyiladi. Hozirgi paytda arifmetik - mantiqiy qurilma va boshqarish qurilmasini birgalikda protsessor yoki markaziy protsessor deb yuritiladi;

- boshlang'ich ma'lumotlarni va ishlash dasturini xotiraga kiritadigan va ish natijasini tashqariga chiqarib beradigan qurilma. Uni kiritish va chiqarish qurilmasi deb yuritiladi.

XX asr boshlariga kelib angliyalik olim A.Tyuring va amerikalik olim E.Post hisoblash mashinasining nazariy asosini yaratgandan keyin hisoblash mashinasi asri boshlandi.

1930 yili amerikalik olim X.Atanosov va K.Berrilar elektron xotira, qo'shish va ayirish qurilmalaridan iborat elektron hisoblash mashinasini yaratdilar.

1937 yili amerikalik olim X.Atanosov hisoblash mashinasi sanoq tizimi uchun ikkilik sanoq tizimini ishlatish g'oyasini berdi va bu yo'nalishda bir necha patentlar ham oldi.

1941 yilda olmon muhandisi K.So'ze Ch.Bebbidj g'oyasi bo'yicha birinchi hisoblash mashinasini yaratdi.

1943 yilda Ch.Bebbidj g'oyasi bo'yicha amerikalik G.Ayken elektro-mexanik relelar yordamida "MARK-1" nomli analitik hisoblash mashinasini yaratdi. 1943 yildan boshlab Amerikada bir guruh mutaxassislar shu g'oya bo'yicha relelar o'rniga elektron lampalardan foydalanib hisoblash mashinasini yaratishga kirishdilar. Ularning yaratgan mashinalari "MARK-1" mashinasidan ming martotaba tez ishlar edi.

1945 yili Germaniyada K.So'ze tomonidan "S-4", 1949-51 yillarda sobiq SSSR da S.Lebedev rahbarligida "MESM" va 1950 yili Angliyada "AKE" kompyuteri yaratildi.

Hisoblash mashinasi - kompyuterlarning ishlash prinsipini umumiy holda tushunarli va sodda qilib bergan olim mashhur Djon fon Neymandir. Odatda bu prinsipni fon Neyman prinsipi deb ham yuritiladi. U ikki prinsipdan iborat:

- hamma kerakli ma'lumotlarni va masalani yechish dasturlarini yagona xotirada saqlab turish;

- kompyuterni dastur yordamida boshqarish.

Hozirgi zamon kompyuterlarining tuzilishi boshqacharoq bo'lib arifmetik-mantiqiy qurilma bilan boshqarish qurilmalari birgalikda markaziy protsessor qurilmasi deb yuritiladi. Ishlash prinsipida ham farqi bo'lib, bir nechta protsessorlar bilan bir vaqtda bir qancha ma'lumotlarni parallel qayta ishlash mumkin. Dastur bilan ishlash davomida zarur bo'lganda uning ishini to'xtatib turib, zarur ishni bajarib, yana oldingi dastur ishini davom ettirishi mumkin.

Kompyuter avlodlari

Kompyuterlar o'zining elementlar bazalari bo'yicha avlodlarga ajratilgan.

I avlod (1945 - 1956 yillar) kompyuterlari elementlar bazalari elektron lampalar ekanligi bilan xarakterlanadi. Bu avlod mashinalari katta zallarni egallagan holda, yuzlab kilovatt elektr energiya sarf qilar va tonnalab og'irlikka ega bo'lib, sekundiga 1-2 ming amal bajarar, xotirasining hajmi 1-2 ming so'zni (ma'lumotni) saqlashga qodir edi. Bu avlod mashinalariga "Ural-1", "Ural-2", "BESM-1", "BESM-2", "M-1", "M-2", "M-20" kabi mashinalarni misol qilib keltirish mumkin.

II avlod (1957 - 1968 yillar) kompyuterlari elementlar bazalari tranzistorlardan iborat edi, tezkorligi sekundiga 10-20 ming amal bajarish, xotirasining hajmi 4-8 ming so'zni saqlashga qodir edi. Ikkinchi avlod kompyuterlari hisoblash ishidan ishlab chiqarish jarayonlarini boshqarish, iqtisodiy masalalarni yechish, harflar bilan ishlay olish "qobiliyati"ga ham ega bo'ldi. Bu avlod mashinalariga "BESM-3", "BESM-4", "Ural-16", "Minsk-22", IBM -608, "BESM-6"ni misol qilib keltirish mumkin.

III avlod (1969 - 1980 yillar) kompyuterlarining elementlar bazalari integral sxemalardan iborat bo'lib, tezkorligi sekundiga 10 mingdan boshlab, shu avlodning eng oxirgi mashinalari 2-2.5 million amal bajarishgacha yetdi. Xotirasining hajmi ham 8-10 ming baytdan (bu avlod xotira o'lchami xalqaro o'lcham baytlarda beriladigan bo'lgan) 8 million baytlargacha yetdi. Bu avlod mashinalariga YaS (yagona seriya) kompyuterlari - "YaS-1010", "YaS-1020", "YaS-1030", "YaS-1035", "YaS-1050", "YaS-1060", "YaS-66" larni misol qilib ko'rsatish mumkin.

IV avlod (1981 - 1990 yillar) kompyuterlarining elementlar bazalari katta integgal sxemalar (KIS) dan iborat. Ularning tezkorligi sekundiga 6,5 million amal bajarishgacha yetdi, xotirasinig hajmi 64 Mega baytgacha kengaydi. Bu avlod mashinalariga Super EHMLar, "Elbrus" 1-KB, "IBM PC" kabi kompyuterlarni ko'rsatish mumkin.

V avlod (1990 yillardan boshlangan) kompyuterlarining elementlar bazalarini o'ta katta integral sxemalar (UKIS) tashkil qiladi. Bu avlod kompyuterlari hozirgi zamonda keng qo'llaniladi. Bu avlod kompyuterlari elektron va yorug'lik nurlari energiyasidan foydalanishga, tuzilishi esa lazer texnikasiga, nurlanuvchi diodlarga asoslangan. Amal bajarish tezligi sekundiga 1 milliardgacha, xotirasinig hajmi 10 milliondan 3-4 milliard (Gbayt) baytgacha kengaydi.

Shaxsiy kompyuterlarining yaratilishi texnikada revolyusion xarakterga ega bo'ldiki, ular ommabop hisoblash mashinalariga aylanib qoldi.

Hozirgi paytda ishlab chiqarish va kundalik hayotda dunyoda 100 millionlab shaxsiy kompyuterlar ishlatilmoqda.

Kompyuter – inglizcha so'z bo'lib, hisoblovchi degani. Uning asosiy vazifasi turli ma'lumotlarni qayta ishlashdan iboratdir. Kompyuterlarning turli xillari mavjud – raqamli, analogli (uzluksiz), maxsuslashtirilgan va h.k. Raqamli kompyuterlardan foydalanish o'zining osonligi, bajariladigan amallarning universalligi va aniqliligi kabi ko'rsatkichlari bilan ajralib turadi.

Hozirda rivojlangan mamlakatlarda kompyuterlarning 5 guruhi keng qo'llaniladi:

1. Super kompyuterlar – yadro sinovlarini va qurollarini modellashtirishda qo'llaniladi. Ular sekundiga 10 trillion amal bajaradi.

2. Katta kompyuterlar (Manfreame Computer) - fan va texnikaning turli sohalariga oid masalalarni yechishga mo'ljallangan. Uning tezligi va xotirasi super kompyuterlarnikidan pastroq. Masalan: AQShda CRAY, Yaponiyada – M1800, Rossiyada ES-9000.

3. Mini kompyuterlar – asosan harbiy maqsadlarda ishlatiladi. Hajmi va tezligi jihatidan katta kompyuterlardan biroz pastroq. Shaxsiy kompyuterlarga o'xshash.

4. Shaxsiy kompyuterlar – jamiyatimizning turli sohalarida ishlatiladi. Hozirgi kunda Pentium toifasidagi protsessorli kompyuterlar keng tarqalgan. Ularning tezligi 750; 1000; 3000 MGS, operativ xotirasi 64; 128; 256 MB ... 4 GB gacha, doimiy xotirasi 40; 80; 120 GB ... 1 TB gacha bo'ladi. IBM, Compaq, HP, Toshiba, Apple, Siemens va h.k. firmalaridan chiqqan kompyuterlar “oq yasalgan” deyiladi. Malayziya, Xitoy, Tayland kabi mamlakatlarda ishlab chiqarilgan kompyuterlar “sariq yasalgan” deyiladi.

5. Noutbuk kompyuterlar (Notebook) - hajmi ancha kichik bo'lgani bilan bajaradigan amallarining soni va xotirasi jihatidan shaxsiy kompyuterlar bilan tenglashmoqda.

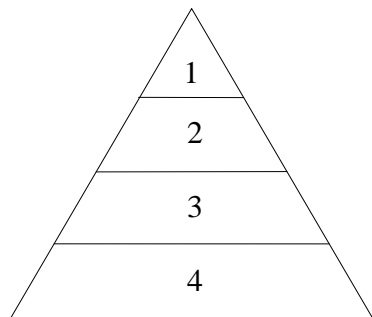
XX asrning 40-yillarida 3 ta yirik davlatlarda elektron hisoblash mashinalarining (EHM) yaratish ustida ish boshlangan. Bular: SSSR, AQSh va Angliya. 1-EHM 1943 yili Angliyada ingliz matematigi va injeneri Alan Tyuring tomonidan yaratilgan. Lekin unchalik shuxrat qozona olmadi. 1945 yil AQShda fizik va matematik fon Neyman tomonidan 1-arifmetik EHM yaratilgan (ENIAC). 1946 yili SSSRda akademik Lebedev boshchiligida EHM yaratilgan. Hozirgi shaxsiy kompyuterlarning arxitekturasi fon Neyman prinsiplari asosiga tayanadi. Ular quyidagilar:

1. Arifmetik – logik (mantiqiy) qurilmaning mavjudligi (ALU);
2. Boshqaruv qurilmasining mavjudligi;
3. Xotiraning mavjudligi;
4. Axborotlarni kiritish - chiqarish qurilmalarining mavjudligi.

Arifmetik-logik qurilma va boshqaruv qurilmasi birgalashib, markaziy protsessorni (MP) tashkil etadi va u kompyuterning asosiy qurilmasi deb hisoblanadi. MP sistemani shinalar orqali tekshirib va boshqarib turadi. Shinalar 2 xil bo'ladi: adres shinasi va ma'lumotlar shinasi. Adres shinasi xotira yacheykasini yoki kiritish -

chiqarish portlarini tanlaydi. Ma'lumotlar shinasi esa MPga kattaliklarni uzatish va undan olish vazifasini bajaradi.

Kompyuterning tez ishlashi uchun xotirani bir necha darajaga bo'lish zarur: juda tez ishlaydigan va sekinroq ishlaydigan:



- 1- MP registrlari
- 2- Kesh xotira
- 3- Operativ xotira
- 4- Doimiy xotira

Registrlar – juda tez ishlaydigan, lekin kam hajmga ega bo'lgan xotira (1 ta registrning hajmi 8, 16, 32 va h.k. bit bo'lishi mumkin). Registrlarga qarab kompyuterlar ham farqlanadi. Agar dasturchi assembler tilida ishlayotgan bo'lsa, registrlar ishini boshqarishi mumkin.

Kesh xotira – tez ishlaydigan xotira turiga kirib, hajmi bir necha yuz Kilo baytdan bir necha MB bo'ladi (Masalan: 256 KB - 4 MB). Uning vazifasi operativ xotirada yaqinda ishlatilgan ba'zi axborotlarni, komandalarni vaqtinchalik saqlab turish. Aynan qaysi axborotni saqlash kerakligini sistemaning o'zi hal qiladi. Dasturchi bu jarayonni boshqara olmaydi.

Operativ xotira – RAM (Read / Write Memory – память для записи/чтения) hajmi kichik, tezligi katta bo'ladi (RAM - Random Access Memory).

Doimiy xotira – ROM (Read Only Memory – память только чтения) katta hajmga ega, tezligi esa pastroq bo'ladi.

Kompyuterlar avval tuzilgan dasturlar asosida ishlaydi. O'z navbatida masalani yechish uchun biror dasturlash tilida yozilgan buyruqlar ketma-ketligini tarjimon dasturlar yordamida kompyuter tiliga o'tkazadi. Kompyuter tili 0 va 1 lardan tashkil topgan ma'lum qoidalar asosida yozilgan ketma-ketlikdan iborat. Dastur avval kompyuter xotirasiga kiritiladi va buyruqlar bajarila boshlaydi. Boshqaruv qurilmasi amallar ustidan nazorat o'rnatadi. Amallarni protsessor bajaradi. Odatda kompyuter 2

qismdan tashkil topgan deyiladi: Hardware - qattiq qism, ya'ni texnik vosita - kompyuter, Software – yumshoq qism, ya'ni dasturiy ta'minot.

EHMning xotira yacheykasi 16, 32, 64 ta 2 lik razryadlardan iborat bo'ladi. Bu 2 lik razryad bit deb ataladi va u axborotning eng kichik o'lcham birligi hisoblanadi. 8 bit – 1 bayt bo'ladi. Bayt nomeri adres deyiladi. 2 bayt 1 mashina so'zi (машинное слово) bo'ladi.

Nazorat uchun savollar

№	
1	Kompyuter - informatikaning texnik asosi.
2	Informatika va kompyuterlar.
3	Kompyuterning rivojlanish tarixi.
4	Kompyuterning asosiy va yordamchi qurilmalari.
5	Protssessor, shinalar, kiritish va chiqarish qurilmalari.
6	Axborotni kompyuterda qabul qilish va qayta ishlash
7	ShK arxitekturasi?
8	«Fon-neyman» tamoyiliga asoslangan kompyuter arxitekturasini tavsiflab bering.
9	Registr nima?
10	Protssessor nima va uning vazifasi?
11	Periferik qurilma turlari va ularga misollar keltiring
12	Shaxsiy kompyuter dasturiy ta'minotining tarkibi
13	Kompyuterdagi operatsion tizim tarkibi va vazifalari
14	Asosi tranzistordan tashkil topgan EHM avlodi
15	Monitorning vazifasi va turlari
16	AMQ (arifmetik mantiqiy qurilmaning) vazifasi
17	Kompyuter qurilmalarining ishini nazorat qiluvchi va dastur bajarilishini kuzatuvchi EHMning asosiy qismi.
18	Kompyuter operatsion tizimining qulayliklari: tarmoqqa kompyuterning

	ulanishi hamda bir nechta kompyuterlarning hisoblash resurslarining to'plashi va ularni birgalikdagi ishlashi.
19	Ishchi stansiyalar uchun hisoblanuvchi asosiy operatsion tizim va maynfreym va mini EHMlarda foydalaniladigan operatsion tizim
20	Kompyuterning apparat va dasturiy qismlarini boshqaruvchi tizim
21	Boshqa dasturlarning bajarilishi, foydalanuvchi bilan muloqotni tashkillashtiruvchi, operativ xotirani taqsimlovchi va h.k. ishlarni bajaruvchi dasturlar kompleksi
22	Kesh xotiraning ishlash prinsipi
23	Kompyuterning asosiy ikkita xotirasi
24	Monitorning rangli oynasidagi tasvirni shakllanishi
25	Tashqi xotira vazifasi. Tashqi xotiradagi qurilma xillari
26	Ko'p darajali xotira tashkili nima uchun ishlatiladi?

2-Laboratoriya ishi. Kompyuterning arifmetik asoslari. Sanoq sistemalari.

Ishning maqsadi:

1. Sanoq sistemalari haqida tushuncha.
2. Sonni bir sanoq sistemasidan boshqasiga o'tkazish.
3. Turli sanoq sistemasidagi sonlar ustida amallar bajarish.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarni bajarishi lozim.
2. Hisobot shaklida oldin vazifa, uni bajarishi, natijalar ko'rsatilishi lozim.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. Laboratoriya ishidagi topshiriq raqami.
3. Topshiriqni bajarilishi.
4. Topshiriq natijasi (har xil variantlar uchun).

NAZARIY QISM

SANOQ SISTEMALARI

Informatikada masalani yechish tushunchasi deganda axborotlarni qayta ishlab, natijani oldindan belgilangan ma'lum bir ko'rinishga olib kelish tushuniladi. Informatika fani 3 ta tarkibiy qismdan iborat:

1. Masalani to'g'ri yechib olish uchun kerakli bilim va mahorat (algoritm va usul);
2. EHMda mavjud bo'lgan tarjimon dasturlardan foydalanib, masalaning dasturini tuzish;
3. Dasturi tuzilgan masalani EHMda yechish va natijani olish.

Masalani kompyuterda yechish bir necha bosqichlarni o'z ichiga oladi:

1. Masalaning qo'yilishi va maqsadi;
2. Masalaning matematik ifodasi;

3. Masalani yechish uchun kerakli usulni aniqlash;
4. Yechish algoritmini tuzish;
5. Algoritmga asosan biror algoritmik tilda dastur tuzish;
6. Dasturni kompyuterga kiritish;
7. Dasturda yuzaga kelgan xatoliklarni tuzatish;
8. Olingan natijani izohlash va tahlil qilish.

1-3 bosqichlar mutaxassisning malakasi va bilimiga bog'liqdir. Bu ish ustida mutaxassislar bir necha oy yoki yillab izlanishlari mumkin. Buning uchun kerakli ma'lumotlar va ular orasidagi bog'lanishlar aniq ifodalangan bo'lishi zarur.

Har qanday hisoblash mashinalarining arifmetik asosi bo'lgan sanoq sistemalari bilan tanishamiz. 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 ham bu o'nlik sanoq sistemasi deb ataladi. Sanoq sistemasida sonlarni yozish uchun qo'llaniladigan raqamlar soni ushbu sanoq sistemasining asosi deb yuritiladi. Kundalik hayotimizda qo'llanib kelinayotgan sanoq sistemasining asosi 10 ga teng.

Sanoq sistemalari, ularning tuzilishi va ulardagi amallar haqidagi tushunchalarni ko'rib chiqamiz.

R simvulli alifbo X ning to'g'ri yozilishi va shu alifbodagi sonlarni shu r simvollar orqali qayta ishlovi sanoq sistemasi deyiladi. Sistemadagi r ga asoslangan X soni $(x)_r$ yoki x_r deb belgilanadi.

Har bir sanoq sistemasi – bu, kodlash va dekodlash operatsiyasini bajaradigan sonli qiymatlarni (to'plamlarni) kodlashdir, ya'ni bunda ixtiyoriy sonli qiymatga kodli ko'rinishni topish va ixtiyoriy kodli yozishga unga mos ravishda sonli qiymatni tiklash kerak bo'ladi.

Hamma sanoq sistemasi bir umumiy prinsipga asosan tuziladi: sistemaning asosi r - kattaligi aniqlanadi, x ixtiyoriy soni esa r - og'irlik darajasining kombinatsiyasi ko'rinishida 0 dan n - chi darajaga quyidagicha o'tadi:

$$(x)_{10} = x_n p^n + x_{n-1} p^{n-1} + \dots + x_1 p^1 + x_0 p^0.$$

Informatikada eng ko'p ishlatiladigani bu 10 lik sanoq sistemasidan tashqari:

1) ikkilik $X=\{0,1\}$;

2) sakkizlik $X=\{0,1,2,3,4,5,6,7\}$;

3) o'n oltilik, $X=\{0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F\}$, bu yerdagi simvollar A,B,C,D,E,F o'nlik sanoq sistemasidagi 10,11,12,13,14,15 ni bildiradi.

$$\text{Misol. } 1102_2 = 1 \cdot 2^3 + 1 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0 = 8 + 4 + 1 = 13_{10},$$

$$157_8 = 1 \cdot 8^2 + 5 \cdot 8^1 + 7 \cdot 8^0 = 64 + 40 + 7 = 111_{10},$$

$$A6F_{16} = 10 \cdot 256 + 6 \cdot 16 + 15 \cdot 1 = 2671_{10}.$$

Ko'pincha, sanoq sistemasidagi hamma sonlar(yoki alifbo simvollari) yozuvdagi joylashgan joyiga bog'liq bo'ladi. Bunday sanoq sistemasi pozitsion deyiladi; aks holda nopozitsion deyiladi.

Misol. Nopozitsion sistema – qadimgi rim alifboning sonlardagi ifodasi, joylashgan joyidan qat'iy nazar quyidagi ko'rinishga ega bo'ladi, $X=\{I(1), V(5), X(10), L(50), C(100), D(500), M(1000)\}$. Rim sonlarining ko'rinishi(qavsda – oddiy o'nlik ekvivalentlari): III(3), IV(4), V(5), VI(6), IX(9), XI(11), DCL(650). Bu sistemadagi yozuv ikki yoqlama konkatenatsiya orqali hosil qilinadi, bunda o'ng taraflama konkatenatsiya qo'shib borish orqali, chap taraflama konkatenatsiya olib tashlash orqalidir(masalan, IV va VI). Arifmetik operatsiyalarni bajarish joylashuviga qaramaydi(masalan, XIV+IV=XVIII).

O'nlik sanoq sistemasidagi kasr sonlarga o'tish va quyidagi formula orqali amalga oshiriladi.

$$\text{Misol. } 110,001_2 = 1x2^2 + 1x2^1 + 0x2^0 + 0x2^{-1} + 0x2^{-2} + 1x2^{-3} = 6,125_{10};$$

$$A, B_{16} = Ax16^0 + Bx16^{-1} = 10x1 + 11x0,0625 = 10,6875_{10}.$$

O'nlik sanoq sistemasidan r-sanoq sistemasiga o'tish yo'llari:

1. X sonining butun qismini ajratib olamiz, avval $[X]_{10}$ ni butun qismini r-ga bo'lamiz, so'ngra kasr qismini r-ga bo'lamiz, to shu sondan kichik son qolguncha va hosil bo'lgan qoldiqlari r-sanoq sistemasini hosil qiladi.

2. $[X]_{10}$ sanoq sistemasidagi sonning kasr qismi (mantissasi) ko'paytiriladi, so'ngra hosil bo'lgan r-dagi sonlar ko'paytiriladi, toki 0ga teng bo'lgan mantissa

hosil bo'lguncha, yoki kerakli $[X]_r$ sonlarining keraklisi hosil bo'lguncha. $[X]_r$ ning ko'rinishi birinchi ko'paytirishdan to oxirgi ko'paytirishga hosil bo'lgan sonlarning butun qismidan hosil qilinadi;

3. Natija quyidagi ko'rinishga ega bo'ladi $(X)_r=[X]_p, \{X\}_p$.

Misol. Topilsin: $12,8=?_2$. Yechimi:

1. Butun qismini olib o'tamiz: $12_{10}=1100_2$;

2. Kasr qismini olib o'tamiz: $0,8*2=1,6$; $0,6*2=1,2$; $0,2*2=0,4$;

$0,4*2=0,8$; $0,8_{10}=0,1100110\dots_2$;

3. Natija: $12,8=1100,1100110011\dots_2$.

Misol. Topilsin $29,25_{10}=?_8$. Natija quyidagi ko'rinishga ega: 1) $29_{10}=35_8$;

2) $0,25_{10}=0,2_8$; 3) $29,25_{10}=35,2_8$.

Sistema asosi			
10	2	8	16
0	0	000	0000
1	1	001	0001
2	-	010	0010
3	-	011	0011
4	-	100	0100
5	-	101	0101
6	-	110	0110
7	-	111	0111
8	-	-	1000
9	-	-	1001
10	-	-	1010
11	-	-	1011
12	-	-	1100
13	-	-	1101
14	-	-	1110
15	-	-	1111

Misol. Topilsin $79,26_{10}=?_{16}$. Natijasi: 1) $79_{10}=4F_{16}$; 2) $0,26_{10}=0,40_{16}$; 3) $79,26_{10}=4F,4_{16}$. Kasr qismini olib o'tganimizda verguldan keyingi ikkita son olinadi va qolgani taxminiy butunlab olinadi, chunki aniq olib o'tishni iloji bo'lmaydi. 2-likka 8-likka teskari yo'l bilan, 2-likdan 16-likka teskari yo'l bilan, 8-likdan 16-likka teskari yo'l bilan va yana orqaga qaytiladi. Quyida keltirilgan jadval ishlatiladi:

8-lik sanoq sistemasiga o'tganda sonlar 3 bitga guruhlanadi, 16-likka o'tganda esa 4 bitga guruhlanadi. Agar kerak bo'lsa, 0 qo'shish mumkin (chapdan butun qismiga va o'ngdan mantissaga) yoki tashlab yuborish kerak bo'ladi.

Misol. Bir sanoq sistemasidan ikkinchisiga o'tishni ko'rib chiqamiz:

1. 2-likdan 8-likka o'tish: $101,10111_2=\underline{101}, \underline{101} \underline{110}_2=5,56_8$; $5_8, 5_8 6_8$	2. 8-likdan 2-likka o'tish: $6,24_8=\underline{6}, \underline{2} \underline{4}_8=110,0101_2$; $110_2, 010_2 100_2$
3. 2-likdan 16-likka o'tish: $101,10111_2=\underline{0101}, \underline{1011} \underline{1000}_2=5,B8_{16}$; $5_{16}, 11(B)_{16} 8_{16}$	4. 16-likdan 2-likka o'tish: $1A,F3_{16}=\underline{1} \underline{A}, \underline{F} \underline{3}_{16}=11010,11110011_2$. $0001_2 1010_2 1111_2 0011$

2-lik sanoq sistemasida qo'shish

$0+0=0, 0+1=1, 1+0=1, 1+1=2_{10}=10_2$ (1 yuqori razryadga o'tadi)

2-likda ayirish

$0-0=0, 1-0=1, 1-1=0, 0-1=10-1=1$ (1 yuqori razryaddan olinadi)

2-likda ko'paytirish quyidagicha

$0*0=0, 0*1=0, 1*0=0, 1*1=1$.

2-likda bo'lish

$0:0=\text{aniqmas}, 1:0=\text{aniqmas}, 0:1=0, 1:1=1$.

Sistemadagi sonning r – ga asoslangan qaytish kodi, shu sistemadagi hosil bo'lgan sonni almashtirilganiga, razryaddagi har bir simvol sistemadagi sonning maksimal to'ldirilishiga aytiladi. (ya'ni $r-1$).

Qo'shimcha kod = qaytish kodi+ quyi razryaddagi bir.

Misol.

1. 10011 =ikkilik soni,

01100= ikkilikdagi qaytish kodi,

01101= ikkilikdagi qo'shimcha kod;

2. 457=sakkizlik soni,

321=yordamchi kod;

3. A9=o'n oltilik soni;

57=yordamchi kod.

Yordamchi kod yordamida ayirish: Razryaddagi ayirmadan yordamchi kodni topish va ayiriluvchiga qo'shish. Ayirma natijasi hosil bo'lgan yig'indi bo'ladi, yuqori razryaddan tashqari.

Misol. To'g'ridan to'g'ri ayirish orqali va qo'shish orqali amalni bajaramiz (yordamchi kod orqali):

$$\begin{array}{r} 110110_2 \\ \underline{10101_2} \\ 100001_2 - \text{ayirma} \end{array} \quad \begin{array}{r} +110110_2 \\ \underline{1011_2} - \text{yordamchi kod} \\ 1100001_2 - \text{yig'indi tashlab yuboriladigan son} \end{array}$$

Matematikada butun sonlar va ularning analogi n-razryadli arifmetikada ayniyatlidir (son qiymati bo'yicha) keltirilgan razryad bo'yicha. Bunda insonning fikrlash darajasi va n-razryadli arifmetik sohasi hisobga olinadi:

1. Cheksiz va hisoblanuvchan butun sonlarning ko'rinishi $[-N; +N]$ kesmada keltiriladi, bunda N- arifmetikada keltirilgan maksimal son (bu yerdagi ko'p nuqta n-ga teng bo'lgan umumiy sonlar birligi): $N=(111\dots1)_2$;

2. Cheksiz va hisoblanuvchan aniq sonlarning $(-\infty; +\infty)$ da sonlar o'qida bir tekis va zich joylashuvi, n-razryadli arifmetikada notekis zichlikda joylashuvi (kichik sonlar orqali siqish);

3. Ko'pchilik aniq sonlarda 0 o'z sohasida ko'p sonlarni tashkil etadi, n-razryadli arifmetikada esa 0 ajratilgan holda keltirilgan. Oddiy arifmetika nuqtai nazaridan, masalan, $(-1; 1)$ chegarasida juda zich joylashgan nuqtalar bor, va har qanday tekislikda shu nuqtalar ichidan bir nuqta mavjud bo'ladi. Bunday arifmetikani regulyar arifmetika deyiladi. Mashina arifmetikasi quyidagi xususiyatlarga ega. U regulyar emas – nuqta intervalida 0 ga yaqinlashganda qo'shib ketadi; bundan

tashqari bu intervalda nuqta x “ajratilgan” holda keladi – agar uning ixtiyoriy sohasida $(x-a; x+a)$, bu yerda a -mashina 0 dan oshmaydigan son, bu intervalda boshqa nuqtalar yo‘q (x dan tashqari). Ehtimollar nazariyasi tarafidan qaraganda, sonlarni zichligini taqsimlash regulyar va noregulyar arifmetikada turlicha. Shuningdek, butun va haqiqiy sonlarni zichligini taqsimlash ham xuddi shu arifmetikada. Ko‘pchilik haqiqiy sonlar mashina arifmetikasida (arifmetika razryadida aniqlangan) ratsional sonlar to‘plami sifatida beriladi. Bu to‘plamlarni boshqa xususiyatlari ham bor (masalan, operatsiyalar bilan bog‘langan), lekin yuqorida keltirilganlari – asosiylari.

Sonlarni oddiy va mashina arifmetikasida (n -razryadli) berilishini farqi, kompyuterning “matematik” imkoniyatlarini chegaralaydi, shuningdek, “kompyuterning” imkoniyatlari matematikadagi matematik usul, algoritmlarni kompyuterda ifodalanishini chegaralaydi.

Shuni nazardan qochirmaslik kerakki, nazariy matematikadagi aniqlik tushunchasi – mavhum. Amaliy matematikada ham, u yo‘q joyda ham, hayoliy aniqlik yuzaga kelishi mumkin. Agar ko‘rib chiqilayotgan hisoblash resurslaridagi xatoliklarni kerakli qiymatlarini chegarasidagi kelishuvlari yo‘q bo‘lsa. Masalan, ishlash va vaqtga nisbatan aytiladigan bo‘lsa, bu boshqarish strategiyasi xatoliklardan holi emas. Bu diapazondagi r ga asoslangan n -razryadli sanoq sistemasi $|(x)_p| \leq p^n - 1$ chegarasida yotadi. Kasr sonlarni keltirish uchun bu diapazon yanada kamayadi. Negaki, razryadning bir qismini mantissa orqali ifodalash kerak bo‘ladi. Xuddi shu usulda “sezmaslik sohasi” atalmish arifmetikadagi n -razryadli sonlar formasi keltiriladi.

1937 yilda Konradom Suze bu diapazonni kengaytirish uchun, arifmetikada keltirilgan ikkilik sonlarni, shuningdek keltirilgan sonlarni aniqligini oshirish uchun, keltirilgan sonlar oquvchan, normal holatga keltirilgan formada keltirilgan, x soni quyidagicha ifodalash mumkin:

$$x = m * p^k,$$

bunda m -mantissa soni, k -butun son tartibi, $p^{-1} \leq |m| < 1$.

Ikkita son berilgan bo'lsin, $x = m * p^k$, va $y = n * p^l (k > l)$. U holda operatsiya bajarilish natijasini tekshirish mumkin:

$$X+y=(m+n * p^{l-k}) * p^k,$$

$$x-y=(m-n * p^{l-k}) * p^k,$$

$$x * y=(m * n) * p^{k+l},$$

$$x/y=(m/n) * p^{k-l}.$$

Agar n-razryaddan, sonlar ko'rinishi sifatida olinadigan, m-ikkilik razryadini mantissa ostiga olish, k – tartib osti, bir razryad – son belgisi ostiga va bir razryad – tartib belgisi ostiga o'tsa (masalan, 0-musbat va 1- manfiy), u holda oquvchi vergulli sonlar formasida keltirilganlar diapazoni birdaniga ko'payadi ($m+k+2=n$):

$$-(0.111...1)_2 * (10)_2^{+(111...1)_2} \leq x \leq +(0.111...1)_2 * (10)_2^{+(111...1)_2}$$

(ko'p nuqta k birligini bildiradi).

Quyi chegaradagi musbat sonlardan kichik sonlar va yuqori chegaradagi manfiy sonlardan katta sonlar 0 ga teng deb olinadi va bir-biridan farqlanmaydi. Yuqori chegaradagi musbat sonlardan katta sonlar musbat cheksizlikka tenglashtiriladi (quyi chegaradagi manfiy sonlardan kichigi – manfiy cheksizlikka tenglashtiriladi). Shuning uchun ikki xil kattalikdagi chegaralangan razryadli arifmetik sonlarni taqqoslash aniq bo'lmagan natijalarga olib keladi, shuningdek matematik nuqtai nazardan qaraganda shu sistemadagi bir-biriga teng ikkita sonlar uchun ham. Bunday tasavvur EHM da saqlashni qulayligi, sonni o'zini saqlash emas, balki uni belgisini, mantissasini, tartibini va tartib belgisini va sonlar operatsiyasi obyekt operatsiyasi bilan solishtiriladi. Bunday obyekt operatsiyasi oddiy: belgilarni solishtirish, kattalashtirish, tartibni kichiklashtirish, matissalarni ko'paytirish, normaga keltirish, ya'ni xulosa qilib aytganda, oddiy siljitish, tenglashtirish operatsiyalarini taqqoslash razryadlarini keltiriladi.

Misol. Sanoq sistemasidagi 3-asosli eng katta va eng kichik 5-razryadli butun sonni hisoblab chiqarish kerak. N-razryadli eng katta butun sonni r-asosli sanoq sistemasida quyidagicha yozish mumkin:

$$x_p^{\max} = \sum_{i=0}^{n-1} (p-1)p^i = \sum_{i=0}^{n-1} p^{i+1} - \sum_{i=0}^{n-1} p^i = p^n - 1$$

Eng kichik n-razryadli butun son bu sistemada:

$$x_{\min} = -x^{\max} = 1 - p^n.$$

Xuddi shuningdek, 3 ga asoslangan sanoq sistemasida va 5 razryadli sonlarda quyidagi sonlar diapazonini keltiramiz:

$$-242 = 1 - 3^5 \leq (x)_4 \leq 3^5 - 1 = 242.$$

Formulalarni yanada ixcham ko'rinishga keltirish mumkin. Masalan, ikkilik sistemasi uchun

$$a = \underbrace{111\dots1}_n = 2^n - 1, b = 1 - 2^n$$

Sakkizlik sanoq sistemasida esa bu sonlar

$$a = \underbrace{777\dots7}_n = 8^n - 1, b = 1 - 8^n$$

Bu xildagi sanoq sistemasi formasidagi "qulaysizliklar" "xavfli" vaziyatlarni yuzaga keltirishi mumkin:

1. Agar son yetarlicha oz bo'lsa, masalan, $a=0.12e+00$, u holda a kiritilgan eng kichik intervalda Ixtiyoriy sonda keltirilishi, 0.120000001 yoki 0.199999999 deb, bu holda ham tengsizlikka taqqoslash mumkin emas (haqiqiy sonlarni sezuvchi vergul formasi bilan taxminiy solishtirish xavflidir);

2. Operatsiyalarni bajarish tartibi natijaga ta'sir etishi mumkin, masalan, 4-razryadli arifmetikada fiksirlangan vergul bilan $20.0000+0.0001=20.0001$, lekin bunda $0.2000e+02+0.1000e-05=0.2000e+02$;

3. Katta sonlarni qo'shishda (ko'paytirishda) "tartibni oshib ketishi" vaziyati yuzaga kelishi mumkin yoki kichik sonlarni qo'shishda (ko'paytirishda) "tartibni yo'qolishi" vaziyati yuzaga kelishi mumkin, $0.6000e+39*0.1200e+64=0.9999e+99$ (yoki aniqlanmagan), shuningdek $0.6000e-35*0.0200e-65=0.9999e-99$ (yoki aniqlanmagan), aniq razryadli o'nlik arifmetikaga mos holatda;

4. Suruluvchi vergulli sonlarni qo'shishda (barcha operatsiyalar qo'shish orqali amalga oshiriladi) navbatdagi mantissalarni qo'shish uchun tartibni tenglashtirish yuzaga keladi, darajalarni tenglashtirishda esa kichik razryadlarni

yo'qotish vujudga keladi, masalan, bunday vaziyat bir "katta sonni" ikkinchi "eng kichik songa" qo'shishda yuzaga keladi.

Sonlar sistemasini bir formaga keltirishni ko'p usullari mavjud (ko'pincha sun'iy).

Misol. Faktorial sanoq sistemasida butun sonlar faktoriallarining chiziqli kombinatsiyasi orqali yoziladi, masalan,

$2457=2*3!+4*2!+5*1!+7*0!$ ($0!=1, n!=1*2*3*...*n$). Bu sistema (shartli) pozitsiondir. Negaki $0!=1!=1$, n -razryadli soni ikkita eng kichigi uchun $x=x_n x_{n-1} \dots x_2 x_1 (n > 1)$ bu soni faktoriallarga taqsimlasak $x_2 * 1! + x_1 * 0! = x_1 * 1! + x_2 * 0!$, shuning uchun ham bu razryadlar og'irligi pozitsiyaga bog'liq emas. (shuning uchun $n > 1$ da bu son nopozitsion shartli ravishda hisoblanadi). Faktorial sanoq sistemasidan o'nlik sanoq sistemasiga o'tish formulasi:

$$x = x_n x_{n-1} \dots x_1 = x_n * (n-1)! + x_{n-1} * (n-2)! + \dots + x_1 * 0!$$

Sanoq sistemasining rivojlanish tarixi ancha qiziqarlidir. Bir xil faktlarnigina keltirib o'tamiz. Ilgari hisob kitob qo'l barmoqlari yordamida qilinardi (oldin beshtalikda, so'ngra o'ntalikda). Bir xil mamlakatlarda 12 ga (masalan, Buyuk Britaniyada – 12 shilling) va 20 ga asoslangan (masalan, Fransiya – "quatre-vings" yoki "to'rt-yigirma" ya'ni 80; qadimgi adiglar analogli ravishda hisob bajarishgan: ya'ni "yigirma-uch" - 60) hisob kitob saqlanib qolgan va shunga o'xshashdir.

Umuman olganda sanoq sistemalarini ikkiga pozitsion va pozitsion bo'lmagan turga ajratish mumkin. Agar biror sanoq sistemasida raqamlar qiymati tushish joyiga (pozitsiyasiga) qarab belgilansa u holda bunday sistema pozitsion sanoq sistemasi, aks holda pozitsion bo'lmagan sanoq sistemasi deyiladi.

Masalan, Rim sanoq sistemasidan boshqa sanoq sistemalarining hammasi pozitsion, rim sanoq sistemasi esa pozitsion bo'lmagan sanoq sistemasiga misol bo'ladi. Haqiqatdan, 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 ham 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 belgilanadi. Shuning uchun o'nlik sanoq sistemasi pozitsion sistemadir. Xuddi shunday ikkilik, sakkizlik, o'n oltilik va boshqa sanoq sistemalari ham pozitsion sistemagan misol bo'la oladi.

Asosi o'ndan katta bo'lgan sanoq sistemalarida qaysi raqamlar ishlatiladi degan savol tug'iladi. Masalan, o'n oltilik sanoq sistemasida biz bilgan o'nta raqam yetarli emas, shuning uchun yana 6 raqami kerak bo'ladi. O'n oltilik sanoq sistemasida bitta raqamlar uchun A, B, C, D, E, F belgilarni kiritsak, o'n oltita raqamga ega bo'lamiz: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. "16" soni esa 10 ko'rinishda yoziladi.

Ma'lumki, o'nlik sanoq sistemasining asosi bir-u nol (10) ko'rinishda yozilgan edi. O'n oltilik sanoq sistemasining asosini ham 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_{10} ; $2,37_8$;

$79A_{16}; 11_2$ yozilgan bo'lsa, bu sonlar mos ravishda o'nlik, sakkizlik, o'n oltilik va ikkilik sanoq sistemasining sonlarini ifodalaydi.

Har qanday asosli sanoq sistemasida qisqa yozuvda berilgan sonlarni asos darajalari bo'yicha yoyib yozish mumkin. Masalan, 451 soni $4 \cdot 10^2 + 5 \cdot 10^1 + 1 \cdot 10^0$ kabi yozish mumkin. Shu kabi quyidagilar ham o'rinli:

Shunday qilib, soni biror asosli pozitsion sanoq sistemasida ifodalash uchun bu soning o'sha sistema asosining darajalari bo'yicha yoyilmasining yig'indisi shaklida yozish yetarli. Demak, biror sanoq sistemasining asos darajalari bo'yicha yoyilmasini mos darajalarga ko'tarib, so'ng qo'shib chiqilsa, hosil bo'lgan son o'nlik sanoq sistemasiga o'tib 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;

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

Demak, $101 = 5$.

2-misol. A_9 sonini o'nlik sanoq sistemasiga o'tkazing;

$$A_9 = A \cdot 16^1 + 9 \cdot 16^0 = 10 \cdot 16 + 9 \cdot 1 = 160 + 9 = 169 .$$

Demak, $A_9 = 169$.

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 ham 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 hosil bo'lgan butun qismdagi raqamlar ketma-ketligi bilan ifodalanadi. Ketma-ket ko'paytirish jarayoni kasr qismi nolga teng bo'lguncha davom ettiriladi.

Ko'rib o'tilgan misollarda kasr qism nollarga aylangan. Lekin har qanday sonlarda ham osongina nollar chiqavermaydi. Hosil 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. Bunda birinchi bosqichda asos darajalari bo'yicha yoyib chiqish, keyingisida ketma-ket bo'lish (butun son) yoki ketma-ket ko'paytirish (kasr son) 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 ham ikkilik va o'n oltilik sanoq sistemi orasidagi bog'lanishni amalga oshirish mumkin. Haqiqatan, to'rtta raqamdan iborat ikkilik sanoq sistemasidagi eng katta son

$$1111 = 1*2 + 1*2 + 1*2 + 1*2 = 8 + 4 + 2 + 1 = 15 \text{ kabi bo'ladi.}$$

Misol. Tetrada yordamida 101010,1101 sonini o'n oltilik sanoq sistemasiga o'tkazing.

Yechish: Buning uchun verguldan boshlab butun qismini chapga qarab, kasr qismini o'ngga 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

0010 1010 1101

---- ----, ----- = 3A,D

3 A D

Axborotlarni raqamlar orqali ifodalash. Axborotning miqdori

Axborotni ma'lum qoida, qonun va belgilar asosida qayta ifodalash kodlash deb ataladi. Qadimda kodlash maxfiy yozuv uchun foydalanilgan. Rim imperatori Yuliy Sezar begonalar davlat ahamiyatiga ega taalluqli ma'lumotlarni o'qiy olmasliklari uchun shartli belgilardan foydalangan. Uning shartli belgisi bo'yicha alifbo aniq sondagi harfga o'ngga yoki chapga surilar edi.

Masalan, biri o'zgarmagan, ikkinchisi bir harfga chapga surilgan ikki qator o'zbekcha harflarni yozaylik:

ABVGDEYOJZIYKLMNOPRSTUFXSCHSH'EYUYAUKGXBGDEYOJZI
YKLMNOPRSTUFXSCHSH'EYUYAO'QG'H

U holda bunday usul bilan "PAXTA" so'zi "RBSUB" ko'rinishda maxfiylashtirilishi mumkin.

Huddi shunga o'xshash kodlashning boshqa usulini ko'rish mumkin. Masalan, alifbo harflarni mos raqamlarga mos qo'yib kodlash mumkin: "a" harfini 1, "b" harfini 2, "v" harfini 3 va h.k. Shu kabi davom etib, "x" harfini 35 soni bilan kodlaylik. U holda, bunday kodlarda "Paxta" so'zini 17; 1; 23; 20; 1; kabi raqamlar ketma – ketligida yozish mumkin, bu usul eng sodda kodlashdir.

Eski telegrafda, masalan axborot Morze alifbosi bilan, ya'ni nuqta va tirelar ketma-ketligi ko'rinishida kodlashtirishlar va uzatilar edi. Masalan, Morze alifbosida STOP so'zi kabi yozilishi mumkin. Kompyuter ixtiyoriy harfni "tanishi" uchun uning

xotirasida harflar har xil usulda yozilgan bo'lishi mumkin. Kompyuter ixtiyoriy harfni" tanishi" uchun uning xotirasida harflar har xil usulda yozilgan bo'lishi kerak.

Shuning uchun uning qo'lingizdagi darslikdagi matn harflarini kompyuter tanishi uchun uning xotirasida harf va belgilarning taxminan 2 ming xil ko'rinishlarini saqlash kerak. Bu juda mushkul va qimmatga tushadigan ish. Bu jarayonni soddalashtirish uchun barcha harflarni 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 raqamlari bilan almashtirish mumkin. Shu yo'sinda tinish belgilarini ham raqamlar orqali kodlash imkoniyati bo'ladi. Masalan, nuqtani 36, vergulni 37 bilan va h.k. Tabiiyki, mashina raqamlari emas, balki raqamlarni ifodalovchi signallar farq qiladi. Xullas, kodlash murakkab tushunchani hammasi bo'lib signalning ikki qiymati bilan (magnitlangan yoki magnitlanmagan, manbaga ulangan yoki ulanmagan, yuqori yoki past kuchlanishi va h.k) ifodalashdir. Bu holatning birinchisini 0 raqami bilan, ikkinchisini esa 1 raqami bilan belgilash qabul qilingan bo'lib, axborotni ikkilikda kodlash nomini olgan. Bunda har bir murakkab tushuncha, ikkilik belgilari ketma-ketligida ifodalanadi. Shunday qilib, quyidagilar bajariladi:

-o'nlik raqamlarini ikkilikda (binarli) kodlash (I K):

-alifbo belgilarini ikkilikda kodlash (axborot almashishning alifboli standart kodi - AASK)

Kodlar ikki: tekis va tekis bo'lmagan turda bo'lishi mumkin. Tekis ikkilik belgilariga ega.

Tekis bo'lmagan kodga Morze alifbosi misol bo'la oladi, chunki unda har bir harf va raqamga uzun va qisqa signallarning ikkilik ketma-ketligi mos keladi. Masalan, E harfiga birgina nuqta mos kelsa, R harfi uchun to'rtta tire mos keladi.

Hisoblash texnikasida odatda tekis kodlarda foydalaniladi. Shular qatoriga axborotlarni kiritish va chiqarish uchun EHMda foydalaniladigan axborot almashinish kodi AAK-8; ikkilik axborot almashinish kodi- IAAK va boshqalarni kiritish mumkin. Ko'pgina zamonaviy kompyuterlarda har bir belgiga 8 bitlik (1 bayt) ketma-ketlik mos qo'yiladi. 8 ta 0 va birlardan tashkil topgan turli ketma-ketliklar jami $2^8=256$ ta bo'lib, ular 256 xil turli belgilarni kodlash mumkin. Masalan, lotin, rus alifbosining katta va kichik harflari, raqamlar, tinish belgilari va boshqa

belgilarini kodlash imkonini beradi. Bayt va belgilarning mosligi, ya'ni har bir kodga mos belgi jadvalda ko'rsatiladi. MDH davlatlarida keng tarqalgan harf raqamli kodlashning AAK-8 (8-xonalik) jadvalni keltiramiz.

O'zbek alifbosi harflarining kodlari lotin alifbosi harflarinikidan farq qiladi. Masalan, o'zbekcha katta "I" harfi 11 101 001, "L" harfi 11 101 100 kodlariga ega.

Nol va birlar ketma-ketligi bilan grafik axborotlarni ham kodlash mumkin. Gazetadagi rasmga diqqat bilan razm solsangiz u mayda nuqtalardan tashkil topganligini ko'rasiz. Turli poligrafiya uskunalarida bu nuqtalarining zichligi turlicha bo'ladi. Ko'pchilik gazetalardagi rasmlarda bir santimetrlik uzunlikda 24 ta nuqta bo'ladi, yani 10x10 sm li rasm taxminan 60 ming nuqtadan iborat. Agar bular faqat oq va qora nuqtalardan iborat bo'lsa, u holda ularning har birini bir bit bilan kodlasa bo'ladi. Agar nuqtalar har xil bo'lsa, u holda bitta nuqtaga bir bit yetarli bo'lmaydi. Ikki bit bilan nuqtaning 4 xil rangini: 00 - oq, 01 - och kul rang, 10 - to'q kul rang, 11 - qora rangni kodlashi mumkin. Uch bit sakkiz xil rangni, 4 bit 16 xil rangni kodlash imkonini beradi va h.k.

Shuningdek, ovozni ham kodlash mumkin. Musiqaga yozilgan notalar ovozni kodlashning turlaridan biridir. Nota belgilariga raqamlarni mos keltirib, ovozni bitlar orqali ifodalash mumkin.

Kompyuter axborotni faqat kodlashtirilgan ko'rinishda qayta ishlaydi. Unga kiritilgan ma'lumotlar xotira qurilmasiga joylashtiriladi. Shuni eslatish zarurki, axborotning mazmuni, uni uzatuvchilarning turi va tashuvchilaridan ma'lumotlarni o'qiydigan kiritish qurilmasining ko'rinishiga bog'liq bo'lmagan holda, ular xotiraga nol va birlar ketma-ketligida yozilishi ma'lum.

Yuqorida eslatilganidek, mashina xotirasi katakchalarga ajratilgan. Katakchada aniq uzunlikdagi har qanday ikkilik raqamlari ketma-ketligi saqlanishi mumkin. Bu ketma-ketlik mashina so'zi deb ataladi. Mashina so'zining uzunligi kompyuterning tuzilishi bilan aniqlanadi. Masalan, kompyuterning xotira katakchasi 24 ikkilik belgidan tashkil topgan mashina so'zini saqlashi mumkin.

Sonlar, belgilar va ko'rsatmalar ham mashina so'zi yordamida ifodalanib qo'llanilishi mumkin. Kompyuter uchun har bir son maxsus ko'rinishdagi so'zdir.

Masalan, haqiqiy sonni tasvirlash uchun uning ishorasi, butun qismi va kasr qismlari ko'rsatilishi kerak, Shuning uchun EHM dastlab mashina so'zining yuqorida sanab o'tilgan xarakteristikalarini aniqlaydi, so'ngra nol va birlar ketma-ketligini aniqlab beradi.

Kompyuterlarda sonlarni tasvirlashning ikki: qo'zg'almas va qo'zg'aluvchi vergulli usulidan foydalaniladi.

Sonlarni qo'zg'almas vergulli tasvirlash. Sonni qo'zg'almas vergulli tasvirlash uchun kompyuter xotirasining katakchasi ishora va raqamlarga mo'ljallangan xonalarga ajratiladi. Katakchanning xonalari odatda, chapdan o'ngga tomon raqamlar bilan tartiblanadi. Katakchanning bitta xonasiga sonning bir xonasi mos keladi. Bunday sonning butun va kasr qismini ajratadigan vergulning o'rnini oldindan belgilanadi.

Qo'zg'almas vergulli tasvirlangan sonlar ustida amallar juda sodda bajariladi, chunki vergulning o'rnini o'zgartmaydi. Shuning uchun xonalardagi raqamlarni mos ravishda qo'shib qo'yish yetarli.

Bu usulning kamchiligi, ishlatiladigan sonlarning chegaralanganligidadir. Haqiqatan EHM 24 xonali xotira katakchasiga ega bo'lib vergul 10-xonadan keyin qo'yiladigan bo'lsa, u holda xotira katakchasidagi absolyut qiymati bo'yicha eng katta son.

Sonlarni qo'zg'aluvchi vergulli tasvirlash. Q asosli sanoq sistemasidagi ixtiyoriy a soni ($a \neq 0$) bu usulda quyidagicha tasvirlanadi: $a = M \cdot Q$, bu yerda M - a sonining mantissasi deyiladi va u musbat to'g'ri kasrdan iborat, r - a sonining tartibi deyiladi va u butun son. Q - sanoq sistemasining asosi.

Bu usulda tasvirlashning kamchiliklari sonni tasvirlashda belgilar sonining ko'payib ketishi va shu bilan mos holda arifmetik amallarni bajarish jarayonining murakkablashib yuborishidan iborat.

Xotira katakchasida sonlarni tasvirlashning ikki usulini ko'rib chiqdik. Shuni aytish kerakki, kompyuterda faqat sonlarni emas, balki turli belgilarni ham tasvirlab ishlatish mumkin. Bunday belgilar ham xotira katakchalarida ularda mos ikkilik kodlari orqali tasvirlanadi.

Kompyuterda axborotlarning ko'rinishi:

Yuqorida aytilganidek, kompyuter faqat sonli ko'rinishdagi ma'lumotlarni qayta ishlay oladi. Har qanday boshqa ma'lumotlarni (tasvir, tovush, harf, raqam, har xil tinish belgilari va h.k.) kompyuterda qayta ishlash uchun ularni sonli ko'rinishda tasvirlash kerak.

Kompyuterda sanoq tizimi uchun ikkilik sanoq tizimi olingan. Shuning uchun har qanday ma'lumotni kompyuterda qayta ishlanadigan bo'lsa, uning ikkilik sanoq tizimida ko'rinishi bilan ish olib borish "bit" deb (yuritiladi) olinadi. 8 ta "bit" ni bir bayt deb nomlangan.

Bitta belgi bir baytda ifodalanadi. Hozirgi paytda kompyuterda ishlatiladigan belgilar soni 256 ta. Bularga 10 ta arab raqamlari, 26 ta lotin katta va kichik harflari, katta va kichik rus harflari, har xil arifmetik va tinish belgilari hamda maxsus belgilar kiradi.

1024 baytni bir kilobayt (1K bayt);

1024 K baytni bir megabayt (1M bayt);

1024 M baytni bir gigabayt (1G bayt) deb ataladi

Nazorat savollari

1. Mashina so'zi nima?
2. Sonlarni tasvirlashning qanday usullari mavjud?
3. Sanoq sistemasining asosi nima?
4. Ixtiyoriy asosli sanoq sistemasidagi sonni yoyib hisoblansa, qanday asosli sanoq sistemalaridagi son hosil bo'ladi?
5. Ixtiyoriy sanoq sistemasida berilgan butun son o'nlik sanoq sistemasiga qanday o'tkaziladi?
6. Ixtiyoriy sanoq sistemasida berilgan kasr son o'nlik sanoq sistemasiga qanday o'tkaziladi?

BAJARISH UCHUN TOPSHIRIQLAR

№	Topshiriq	
1.	a)	b)
	$\begin{array}{r} + 11111111 \\ + \underline{10111011} \\ \hline * 10011011 \\ \quad \underline{111} \end{array}$ $\begin{array}{r} - 10010110 \\ - \underline{1111111} \\ \hline 110010 \mid 101 \\ \quad \quad \quad \underline{\quad} \end{array}$	$\begin{array}{r} + 105771 \\ + \underline{72430} \\ \hline * 375 \\ \quad \underline{12} \end{array}$ $\begin{array}{r} - 105001 \\ - \underline{72430} \\ \hline 510 \mid 101 \\ \quad \quad \quad \underline{\quad} \end{array}$
2.	a)	b)
	$\begin{array}{r} + 10110111 \\ + \underline{11011001} \\ \hline * 101101 \\ \quad \underline{111} \end{array}$ $\begin{array}{r} - 11000110 \\ - \underline{10111101} \\ \hline 10101111 \mid 111 \\ \quad \quad \quad \underline{\quad} \end{array}$	$\begin{array}{r} + 10767 \\ + \underline{4335} \\ \hline * 556 \\ \quad \underline{31} \end{array}$ $\begin{array}{r} - 10007 \\ - \underline{4335} \\ \hline 7562 \mid 16 \\ \quad \quad \quad \underline{\quad} \end{array}$
3.	a)	b)
	$\begin{array}{r} + 111011011 \\ + \underline{1010011011} \\ \hline * 10110111 \\ \quad \underline{101} \end{array}$ $\begin{array}{r} - 11011000 \\ - \underline{10111111} \\ \hline 100100111 \mid 111 \\ \quad \quad \quad \underline{\quad} \end{array}$	$\begin{array}{r} + 60737 \\ + \underline{21645} \\ \hline * 117 \\ \quad \underline{24} \end{array}$ $\begin{array}{r} - 60005 \\ - \underline{21645} \\ \hline 2056 \mid 16 \\ \quad \quad \quad \underline{\quad} \end{array}$
4.	a)	b)
	$\begin{array}{r} + 11011011 \\ + \underline{11001100} \\ \hline * 10110111 \\ \quad \underline{111} \end{array}$ $\begin{array}{r} - 11011000 \\ - \underline{10111111} \\ \hline 10010011 \mid 111 \\ \quad \quad \quad \underline{\quad} \end{array}$	$\begin{array}{r} + 71462 \\ + \underline{3576} \\ \hline * 506 \\ \quad \underline{23} \end{array}$ $\begin{array}{r} - 71001 \\ - \underline{3576} \\ \hline 5442 \mid 12 \\ \quad \quad \quad \underline{\quad} \end{array}$
5.	a)	b)
	$\begin{array}{r} + 10101010 \\ + \underline{11111111} \\ \hline * 1011011 \\ \quad \underline{101} \end{array}$ $\begin{array}{r} - 11001010 \\ - \underline{1111111} \\ \hline 101010111 \mid 111 \\ \quad \quad \quad \underline{\quad} \end{array}$	$\begin{array}{r} + 46173 \\ + \underline{4205} \\ \hline * 225 \\ \quad \underline{47} \end{array}$ $\begin{array}{r} - 46003 \\ - \underline{4205} \\ \hline 7175 \mid 11 \\ \quad \quad \quad \underline{\quad} \end{array}$
6.	a)	b)
	$\begin{array}{r} + 11100011 \\ + \underline{11100111} \\ \hline * 11011011 \\ \quad \underline{101} \end{array}$ $\begin{array}{r} - 11010100 \\ - \underline{1101111} \\ \hline 10101011 \mid 101 \\ \quad \quad \quad \underline{\quad} \end{array}$	$\begin{array}{r} + 34546 \\ + \underline{2177} \\ \hline * 225 \\ \quad \underline{47} \end{array}$ $\begin{array}{r} - 37006 \\ - \underline{2177} \\ \hline 6750 \mid 12 \\ \quad \quad \quad \underline{\quad} \end{array}$

7.	a)	b)
	$ \begin{array}{r} +11111000 \\ \hline 10101111 \end{array} $ $ \begin{array}{r} -10011100 \\ \hline 1110011 \end{array} $ $ \begin{array}{r} *10011100 \\ \hline 111 \end{array} $ $ \begin{array}{r} 10000111 101 \\ \hline \end{array} $	$ \begin{array}{r} +71463 \\ \hline 7325 \end{array} $ $ \begin{array}{r} -71003 \\ \hline 7325 \end{array} $ $ \begin{array}{r} *107 \\ \hline 56 \end{array} $ $ \begin{array}{r} 3735 13 \\ \hline \end{array} $
8.	a)	b)
	$ \begin{array}{r} +10111110 \\ \hline 10111110 \end{array} $ $ \begin{array}{r} -1101101 \\ \hline 1011010 \end{array} $ $ \begin{array}{r} *10010110 \\ \hline 111 \end{array} $ $ \begin{array}{r} 11000100 111 \\ \hline \end{array} $	$ \begin{array}{r} 51742 \\ +4136 \\ \hline \end{array} $ $ \begin{array}{r} -51002 \\ \hline 4132 \end{array} $ $ \begin{array}{r} *351 \\ \hline 11 \end{array} $ $ \begin{array}{r} 3401 13 \\ \hline \end{array} $
9.	a)	b)
	$ \begin{array}{r} +10000110 \\ \hline 11111111 \end{array} $ $ \begin{array}{r} -11101110 \\ \hline 10110110 \end{array} $ $ \begin{array}{r} *11000110 \\ \hline 111 \end{array} $ $ \begin{array}{r} 1110011 101 \\ \hline \end{array} $	$ \begin{array}{r} +75614 \\ \hline 2164 \end{array} $ $ \begin{array}{r} -75004 \\ \hline 2164 \end{array} $ $ \begin{array}{r} *402 \\ \hline 37 \end{array} $ $ \begin{array}{r} 11451 11 \\ \hline \end{array} $
10.	a)	b)
	$ \begin{array}{r} +10010110 \\ \hline 10110111 \end{array} $ $ \begin{array}{r} -11100110 \\ \hline 10011101 \end{array} $ $ \begin{array}{r} *10011100 \\ \hline 111 \end{array} $ $ \begin{array}{r} 1001011 101 \\ \hline \end{array} $	$ \begin{array}{r} 67435 \\ +2135 \\ \hline \end{array} $ $ \begin{array}{r} -67005 \\ \hline 2135 \end{array} $ $ \begin{array}{r} *612 \\ \hline 53 \end{array} $ $ \begin{array}{r} 7175 11 \\ \hline \end{array} $
11.	a)	b)
	$ \begin{array}{r} +11011011 \\ \hline 11001100 \end{array} $ $ \begin{array}{r} -11010100 \\ \hline 1101111 \end{array} $ $ \begin{array}{r} *10101010 \\ \hline 111 \end{array} $ $ \begin{array}{r} 10001110 101 \\ \hline \end{array} $	$ \begin{array}{r} 71462 \\ +3576 \\ \hline \end{array} $ $ \begin{array}{r} -71002 \\ \hline 3576 \end{array} $ $ \begin{array}{r} *415 \\ \hline 65 \end{array} $ $ \begin{array}{r} 6750 11 \\ \hline \end{array} $
12.	a)	b)
	$ \begin{array}{r} +10111011 \\ \hline 11001110 \end{array} $ $ \begin{array}{r} -11100110 \\ \hline 10011101 \end{array} $ $ \begin{array}{r} *10011100 \\ \hline 111 \end{array} $ $ \begin{array}{r} 10101011 111 \\ \hline \end{array} $	$ \begin{array}{r} +43675 \\ \hline 1026 \end{array} $ $ \begin{array}{r} -43005 \\ \hline 1026 \end{array} $ $ \begin{array}{r} *425 \\ \hline 65 \end{array} $ $ \begin{array}{r} 2537 15 \\ \hline \end{array} $

13.	a)	b)
	$\begin{array}{r} + 10000110 \\ \hline + 11111111 \\ \hline \end{array}$ $\begin{array}{r} - 11101110 \\ \hline - 10110110 \\ \hline \end{array}$ $\begin{array}{r} * 11000110 \\ \hline \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 110011 \mid 101 \\ \hline \end{array}$	$\begin{array}{r} + 75614 \\ \hline + 2164 \\ \hline \end{array}$ $\begin{array}{r} - 75004 \\ \hline - 2164 \\ \hline \end{array}$ $\begin{array}{r} * 402 \\ \hline \quad 65 \\ \hline \end{array}$ $\begin{array}{r} 1145 \mid 111 \\ \hline \end{array}$
14.	a)	b)
	$\begin{array}{r} + 10111110 \\ \hline + 10111110 \\ \hline \end{array}$ $\begin{array}{r} - 1101101 \\ \hline - 1011010 \\ \hline \end{array}$ $\begin{array}{r} * 10010110 \\ \hline \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 11000100 \mid 111 \\ \hline \end{array}$	$\begin{array}{r} + 51742 \\ \hline + 4136 \\ \hline \end{array}$ $\begin{array}{r} - 51002 \\ \hline - 4136 \\ \hline \end{array}$ $\begin{array}{r} * 351 \\ \hline \quad 11 \\ \hline \end{array}$ $\begin{array}{r} 3401 \mid 13 \\ \hline \end{array}$
15.	a)	b)
	$\begin{array}{r} + 11111000 \\ \hline + 10101111 \\ \hline \end{array}$ $\begin{array}{r} - 10011100 \\ \hline - 1001101 \\ \hline \end{array}$ $\begin{array}{r} * 10011001 \\ \hline \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 10000111 \mid 101 \\ \hline \end{array}$	$\begin{array}{r} + 71463 \\ \hline + 7325 \\ \hline \end{array}$ $\begin{array}{r} - 71003 \\ \hline - 7325 \\ \hline \end{array}$ $\begin{array}{r} * 107 \\ \hline \quad 56 \\ \hline \end{array}$ $\begin{array}{r} 3735 \mid 13 \\ \hline \end{array}$
16.	a)	b)
	$\begin{array}{r} + 11100011 \\ \hline + 11100111 \\ \hline \end{array}$ $\begin{array}{r} - 11010100 \\ \hline - 1101111 \\ \hline \end{array}$ $\begin{array}{r} * 11000011 \\ \hline \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 100011110 \mid 101 \\ \hline \end{array}$	$\begin{array}{r} + 37546 \\ \hline + 2177 \\ \hline \end{array}$ $\begin{array}{r} - 37006 \\ \hline - 2177 \\ \hline \end{array}$ $\begin{array}{r} * 415 \\ \hline \quad 65 \\ \hline \end{array}$ $\begin{array}{r} 6750 \mid 11 \\ \hline \end{array}$
17.	a)	b)
	$\begin{array}{r} + 10101010 \\ \hline + 11111111 \\ \hline \end{array}$ $\begin{array}{r} - 11001010 \\ \hline - 1111111 \\ \hline \end{array}$ $\begin{array}{r} * 11011011 \\ \hline \quad 101 \\ \hline \end{array}$ $\begin{array}{r} 101010111 \mid 111 \\ \hline \end{array}$	$\begin{array}{r} + 46173 \\ \hline + 4205 \\ \hline \end{array}$ $\begin{array}{r} - 46003 \\ \hline - 4205 \\ \hline \end{array}$ $\begin{array}{r} * 225 \\ \hline \quad 47 \\ \hline \end{array}$ $\begin{array}{r} 7175 \mid 11 \\ \hline \end{array}$
18.	a)	b)
	$\begin{array}{r} + 11011011 \\ \hline + 11001100 \\ \hline \end{array}$ $\begin{array}{r} - 111111000 \\ \hline - 100111111 \\ \hline \end{array}$ $\begin{array}{r} * 10101010 \\ \hline \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 1110011 \mid 101 \\ \hline \end{array}$	$\begin{array}{r} + 71462 \\ \hline + 3576 \\ \hline \end{array}$ $\begin{array}{r} - 71002 \\ \hline - 3576 \\ \hline \end{array}$ $\begin{array}{r} * 502 \\ \hline \quad 23 \\ \hline \end{array}$ $\begin{array}{r} 5442 \mid 12 \\ \hline \end{array}$

19.	a)	b)
	$\begin{array}{r} 1110111011 \\ + 1010011011 \\ \hline \end{array}$ $\begin{array}{r} 10110111 \\ * \quad 101 \\ \hline \end{array}$ $\begin{array}{r} 11011000 \\ - 10111111 \\ \hline \end{array}$ $\begin{array}{r} 10010011 111 \\ \hline \end{array}$	$\begin{array}{r} 60735 \\ + 21645 \\ \hline \end{array}$ $\begin{array}{r} 60005 \\ - 21645 \\ \hline \end{array}$ $\begin{array}{r} 117 \ 2056 16 \\ * \quad 24 \\ \hline \end{array}$
20.	a)	b)
	$\begin{array}{r} 10110111 \\ + 11011001 \\ \hline \end{array}$ $\begin{array}{r} 11000110 \\ - 10111101 \\ \hline \end{array}$ $\begin{array}{r} 1101101 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 10101111 101 \\ \hline \end{array}$	$\begin{array}{r} 10767 \\ + 4335 \\ \hline \end{array}$ $\begin{array}{r} 10007 \\ - 4335 \\ \hline \end{array}$ $\begin{array}{r} 556 \ 7562 16 \\ * \quad 31 \\ \hline \end{array}$
21.	a)	b)
	$\begin{array}{r} 11111111 \\ + 10111011 \\ \hline \end{array}$ $\begin{array}{r} 10010110 \\ - 11111111 \\ \hline \end{array}$ $\begin{array}{r} 10110111 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 110010 101 \\ \hline \end{array}$	$\begin{array}{r} 105771 \\ + 72430 \\ \hline \end{array}$ $\begin{array}{r} 105001 \\ - 72430 \\ \hline \end{array}$ $\begin{array}{r} 375 \ 510 17 \\ * \quad 12 \\ \hline \end{array}$
22.	a)	b)
	$\begin{array}{r} 10110111 \\ + 11011001 \\ \hline \end{array}$ $\begin{array}{r} 11001010 \\ - 11111111 \\ \hline \end{array}$ $\begin{array}{r} 11000110 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 10101011 111 \\ \hline \end{array}$	$\begin{array}{r} 357761 \\ + 2647 \\ \hline \end{array}$ $\begin{array}{r} 35001 \\ - 2647 \\ \hline \end{array}$ $\begin{array}{r} 351 \ 3401 13 \\ * \quad 31 \\ \hline \end{array}$
23.	a)	b)
	$\begin{array}{r} 10111110 \\ + 10111110 \\ \hline \end{array}$ $\begin{array}{r} 111111000 \\ - 100111111 \\ \hline \end{array}$ $\begin{array}{r} 10010110 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 1110011 101 \\ \hline \end{array}$	$\begin{array}{r} 26734 \\ + 4516 \\ \hline \end{array}$ $\begin{array}{r} 26004 \\ - 4516 \\ \hline \end{array}$ $\begin{array}{r} 506 \ 5442 12 \\ * \quad 43 \\ \hline \end{array}$
24.	a)	b)
	$\begin{array}{r} 11100011 \\ + 11100111 \\ \hline \end{array}$ $\begin{array}{r} 11010100 \\ - 11011111 \\ \hline \end{array}$ $\begin{array}{r} 11000011 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 10000111 101 \\ \hline \end{array}$	$\begin{array}{r} 60735 \\ + 21645 \\ \hline \end{array}$ $\begin{array}{r} 60005 \\ - 21645 \\ \hline \end{array}$ $\begin{array}{r} 117 \ 2056 6 \\ * \quad 24 \\ \hline \end{array}$

25.	a)	b)
	$\begin{array}{r} 10010110 \\ + 10110111 \\ \hline \end{array}$ $\begin{array}{r} 10011100 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 11010100 \\ - 1101111 \\ \hline \end{array}$ $10001110 \mid 101$	$\begin{array}{r} 67435 \\ + 2132 \\ \hline \end{array}$ $\begin{array}{r} 67005 \\ - 2135 \\ \hline \end{array}$ $425 \quad 2537 \mid 5$ $* \quad 65$
26.	a)	b)
	$\begin{array}{r} 10101010 \\ + 11111111 \\ \hline \end{array}$ $\begin{array}{r} 11011011 \\ * \quad 101 \\ \hline \end{array}$ $\begin{array}{r} 11001010 \\ - 1111111 \\ \hline \end{array}$ $101010111 \mid 111$	$\begin{array}{r} 71462 \\ + 3576 \\ \hline \end{array}$ $\begin{array}{r} 71001 \\ - 3576 \\ \hline \end{array}$ $506 \quad 5442 \mid 12$ $* \quad 23$
27.	a)	b)
	$\begin{array}{r} 11111111 \\ + 10111011 \\ \hline \end{array}$ $\begin{array}{r} 10110111 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 10010110 \\ - 1111111 \\ \hline \end{array}$ $110010 \mid 101$	$\begin{array}{r} 51742 \\ + 4136 \\ \hline \end{array}$ $\begin{array}{r} 51002 \\ - 4136 \\ \hline \end{array}$ $351 \quad 3401 \mid 13$ $* \quad 11$
28.	a)	b)
	$\begin{array}{r} 10110111 \\ + 11011001 \\ \hline \end{array}$ $\begin{array}{r} 101101 \\ * \quad 11 \\ \hline \end{array}$ $\begin{array}{r} 11000110 \\ - 10111101 \\ \hline \end{array}$ $10101111 \mid 111$	$\begin{array}{r} 357761 \\ + 2647 \\ \hline \end{array}$ $\begin{array}{r} 35001 \\ - 2647 \\ \hline \end{array}$ $601 \quad 3136 \mid 12$ $* \quad 52$
29.	a)	b)
	$\begin{array}{r} 10111011 \\ + 11001110 \\ \hline \end{array}$ $\begin{array}{r} 10011001 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 11100110 \\ - 10011101 \\ \hline \end{array}$ $10000111 \mid 101$	$\begin{array}{r} 46173 \\ + 4205 \\ \hline \end{array}$ $\begin{array}{r} 46003 \\ - 4205 \\ \hline \end{array}$ $225 \quad 7175 \mid 11$ $* \quad 47$
30.	a)	b)
	$\begin{array}{r} 11100011 \\ + 11100111 \\ \hline \end{array}$ $\begin{array}{r} 101101 \\ * \quad 111 \\ \hline \end{array}$ $\begin{array}{r} 11010100 \\ - 1011111 \\ \hline \end{array}$ $10101111 \mid 111$	$\begin{array}{r} 10767 \\ + 4335 \\ \hline \end{array}$ $\begin{array}{r} 10007 \\ - 4335 \\ \hline \end{array}$ $556 \quad 7562 \mid 16$ $* \quad 31$

Bir sanoq sistemasidan boshqa sanoq sistemasiga o'tkazing

N₂	X₂	Y₁₀
1.	100011,01	409,7
2.	110011,01	2041,2
3.	1010110,11	408,6
4.	1011,01	250,3
5.	100001,10	179,8
6.	101101,11	405,1
7.	111111,11	364,3
8.	10001,10	198,1
9.	101100,11	273,1
10.	111110,11	157,3
11.	1011001,11	126,08
12.	100101,11	441,03
13.	101011,10	251,6
14.	101111,11	102,5
15.	1011011,10	205,1
16.	1011011,01	409,6
17.	1011110,01	307,9
18.	101000,11	126,03
19.	110001,01	226,08
20.	111101,11	493,01
21.	1011011,01	199,6
22.	101101,11	375,3
23.	101001,11	266,8
24.	111101,11	399,3
25.	110101,11	181,01
26.	110100,01	411,03
27.	1110111,11	299,06
28.	1101101,011	198,0325
29.	1000111,001	997,1
30.	1001001,11	203,7

3-Laboratoriya ishi. Algoritmning turli tuzilmalari

Ishning maqsadi

1. Algoritm, xossalari, turlari.
2. Algoritmni ifodalash usullari.
3. Chiziqli algoritmlar.
4. Tarmoqlanuvchi algoritmlar.
5. Takrorlanuvchi algoritmlar.

Topshiriq

1. Har bir talaba jurnaldagi tartib raqami bo'yicha vazifalarning algoritmini tuzish.
2. 3-laboratoriya beriladigan barcha vazifalarni bitta hisobot shaklida topshiring.

NAZARIY QISM

Algoritm tushunchasi va uning xossalari

Algoritm so'zi IX asrda yashab ijod etgan jahonda mashhur o'zbek matematigi Al - Xorazmiy nomining lotincha Algoritm o'qilishidan kelib chiqqan.

Algoritm - bu biror masalani yechish uchun mo'ljallangan va bajarilishi kerak bo'lgan aniq harakatlarning chekli ketma-ketligidir.

Algoritmni sof matematik tushuncha deb ta'riflab bo'lmaydi. Nafaqat hayotimizda, balki matematikada ham ayrim jarayonlar taxminan tavsiflanadi. Biror masalani algoritmlashtirish jarayonida uni formallashtirish zarur.

Formallashtirish jarayoni-bu ko'rib chiqilayotgan masalani so'zlar, formulalar, ko'rsatmalar yoki boshqa yo'l bilan aniq algoritm tuzish darajasiga yetkazish. Formallashtirish natijasida masalaning modeli tuziladi. Tuzilgan model asosida masalaning algoritmi bajariladi.

Model -bu obyekt, obyekt majmui yoki biron jarayonni matematik munosabatlari, matn, jadvallar yoki boshqa yo'l bilan formal ta'riflash.

Algoritmning asosiy xossalari sifatida quyidagilarni keltirishimiz mumkin.

1. **Algoritmning uzlukligi (diskretligi)** - bu algoritmning har bir qadami to'la tugallanganidan so'ngina navbatdagi qadaming bajarilishini anglatadi. Ya'ni, uzluklilik xususiyati algoritmning chekli qadamlardan iborat ekanligidan kelib chiqadi.

2. **Algoritmning tushunarligi** - bu algoritmni tuzish vaqtida har bir qadam algoritm bajaruvchining imkoniyatlarini to'liq hisobga olgan holda yozilishi kerak bo'lishini anglatadi.

3. **Algoritmning aniqliligi** - bu uning har bir qadami bir xil ma'noda tushunilib uning o'zidan oldingi qadamlar bilan to'laligicha aniqlanishi demakdir. Algoritmning bu xossasi shundan iboratki, algoritmning har bir qadami aniq ko'rsatilgan bo'lishi kerakligini bildiradi.

4. **Algoritmning umumiyligi** - bu ma'lum bir masalani yechish uchun ishlab chiqilgan algoritmni shunga o'xshash bir qancha masalalarga qo'llash mumkinligidir.

5. **Algoritmning natijaviyligi** - bu algoritmning qadamlar soni chekli bo'lib ushbu qadamlari to'liq bajarilganidan so'ng aniq bir natija olishidir. Bu xossa, oddiy qilib aytganda ,algoritmning so'nngi qadami bo'lishini talab etadi.

Algoritmni ifodalash usullari

Algoritmlar asosan odatda uch xil usulda ifodalanadi:

1. **Algoritmning so'zlar bilan ifodalanishi** har bir ko'rsatma so'zlar orqali buyruq mazmunida beriladi. Shuningdek algoritmning formulalar yordamida ham ifodalanishi matematika, fizika, kimyo, kabi aniq fanlarni yechishda foydalanamiz. Ba'zi hollarda algoritm jadval ko'rinishida ham berilishi mumkin. Misol: 4 xonali matematik jadvallar.

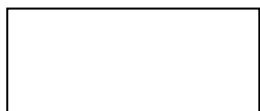
2. **Algoritmni operatorli ifodalanishi.** Bu usulda algoritmdagi har bir harakat bajaruvchi tushunadigan tilda ifodalanishi kerak. Algoritmni so'z orqali ifodalanishining kamchiligi shundaki, algoritm tuzuvchining harakatlarni qisqa va aniq ifodalashiga bog'liq bo'ladi. Shuning uchun odatda algoritm grafik ko'rinishida ifodalanadi.

3. Algoritmning grafik ko'rinishida har bir harakatga ma'lum bir blok moslashtiriladi. Ya'ni grafik tarzida tasvirlangan algoritm ma'lum bir qoida bo'yicha o'zaro bog'langan geometrik shakllardan iborat bo'ladi.

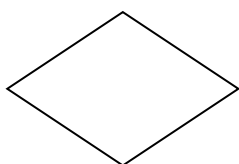
Algoritmning grafik ko'rinishida qo'llaniladigan asosiy geometrik shakllarini ko'rib chiqamiz.



Bu shakl algoritmning boshlanishi va tamom bo'lishini belgilaydi.



To'g'ri to'rtburchak bilan algoritmda bajariladigan hisoblash amallari ko'rsatiladi.



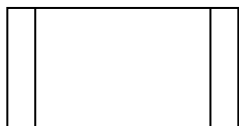
Romb shaklida tekshirilishi kerak bo'ladigan masalaning Sharti yoziladi: Romb Shartining bajarilishi yoki bajarilmasligiga mos ravishda "HA" yoki "YO'Q" ikkita chiqish yo'liga ega.



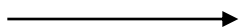
Parallelogramm shakli bilan kiritiladigan kattaliklar va chiqariladigan natijalar ifodalanadi.



Modifikasiya bloki. Bu shakl bilan siklning boshqarish parametri o'zgarishi beriladi.



Standart dastur bloki. Bu shakl bilan qo'llaniladigan standart dasturlar ko'rsatiladi.



Oqim chizig'i. Dastur bloklar orasida aloqalar o'rnatadi.



Qiymatni chiqarish.

Algoritmning grafik ko'rinishi **blok-sxema** deyiladi. Masalaning algoritm blok-sxemasini tayyorlashga ma'lum talablar qo'yiladi. Birinchi navbatda hamma shakllar bir tekisda shakllanishi, bloklarning kattaligi bir o'lchamda chizilishi lozim.

4. Algoritmning operatorli ifodalanishi. Bu usulda algoritmni ifodalash uchun maxsus operatorlardan foydalaniladi. Algoritmning kompyuter tushunadigan biror tildagi ifodasi programma deb ataladi. Kompyuterda ishlatiladigan tillar

algoritmik tillar deyiladi. Algoritmik tildagi harakatlarni ifodalovchi iboralar operatorlar deyiladi. Algoritmik tillar uchga bo'linadi:

1. Yuqori daragadagi algoritmik tillar (Basic, paskal va h.k);
2. Biror EHM turi uchun mo'ljallangan algoritmik tillar (assembler, C, C++);
3. Ma'lum bir sinf masalalarini yechish uchun mo'ljallangan algoritmik tillar (Access, Foxpro).

Algoritmni ifodalash turlari (tuzilmalari). Algoritm asosan uch xil tuzilmadan iborat bo'ladi:

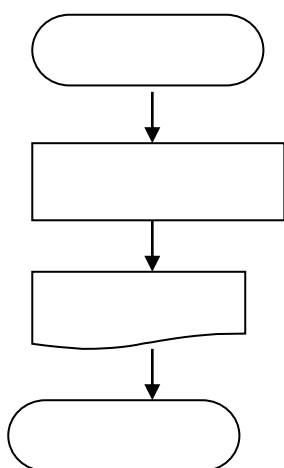
1. Chiziqli tuzilma;
2. Tarmoqlanuvchi tuzilma;
3. Takrorlanuvchi tuzilma.

Algoritmilar chiziqli, tarmoqlangan, takrorlanuvchi turlari asosida tuziladi. Chiziqli, tarmoqlangan va takrorlanuvchi hisoblash jarayonlarining algoritmilarini ishlab chiqamiz.

Chiziqli struktura. Oldindan chiziqli algoritm bilan tanishib chiqamiz.

Harakatlar yozilish ketma-ketligida bajariluvchi tuzilmaga chiziqli tuzilma deyiladi. Chiziqli algoritmni ifodalash uchun ketma-ketlik blokidan foydalanamiz.

Misol: $U=(X-4):(X+4)$ Bu chiziqli algoritmda bloklar qanday joylashgan bo'lsa, xuddi shu tartibda bajariladi.



Keltirilgan algoritmning blok sxemasida birinchi blokda X o'zgaruvchi kiritiladi.

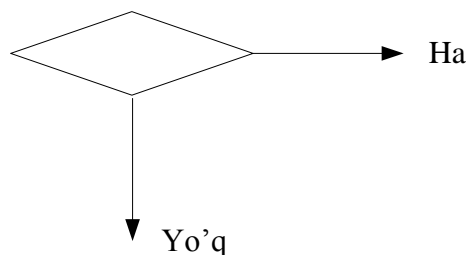
Ikkinchi blokda misol shartidagi formula yechilishi natijasida olingan qiymat U ga o'zlashtiriladi.

Uchinchi blokda uning topib olingan qiymati chiqariladi .

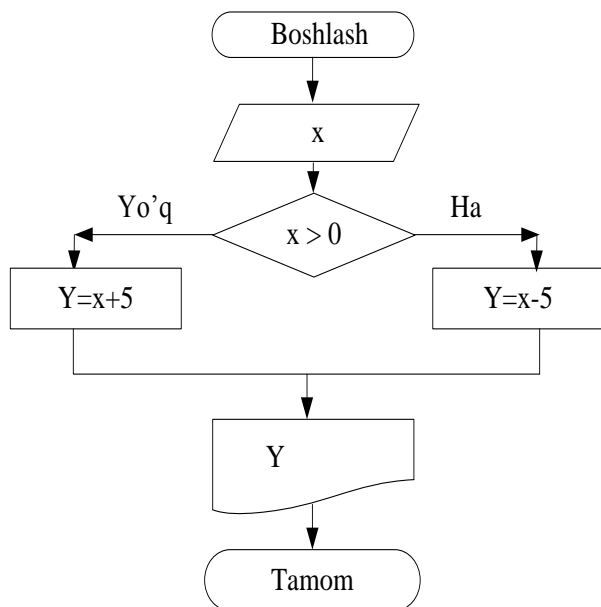
To'rtinchi blok esa dastur tugallanishini bildiradi.

Tarmoqlanuvchi tuzilma

Agar masalani yechish usuli biror berilgan shartning bajarilishiga qarab turli harakatlarning ketma-ketligiga tarmoqlangan bo'lsa va masalani yechish jarayonida bu tarmoqlardan faqat bittasi bajarilsa, bunday hisoblash jarayonlari tarmoqlanuvchi hisoblash jarayonlari deyiladi. Tarmoqlanuvchi hisoblash jarayonlarini algoritmlash uchun tarmoqlanuvchi tuzilma ishlatiladi. Tuzilma ko'rinishi quyidagicha.



Bu yerda A-Shart, A_1, A_2 harakat yoki harakatlar ketma-ketligi. Bunda A shartning qanoatlangan (Ha) holda A_2 , aks holda (Yo'q) A_1 bajariladi. **Misol:** Shartning bajarilishini quyidagi misolda ko'rib chiqamiz .



$$Y = \begin{cases} X - 5 & \text{agar } X > 0 \\ X + 5 & \text{agar } X \leq 0 \end{cases}$$

Agar $X > 0$ Sharti bajarilsa, u holda hisoblash jarayonida 2,3,4,6 bloklar ishlatiladi, aks holda 2,3,5,6 bloklar ishlatiladi. Shunday qilib, ikkita bloklardan faqat bittasi ishlatiladi: 4 yoki 5.

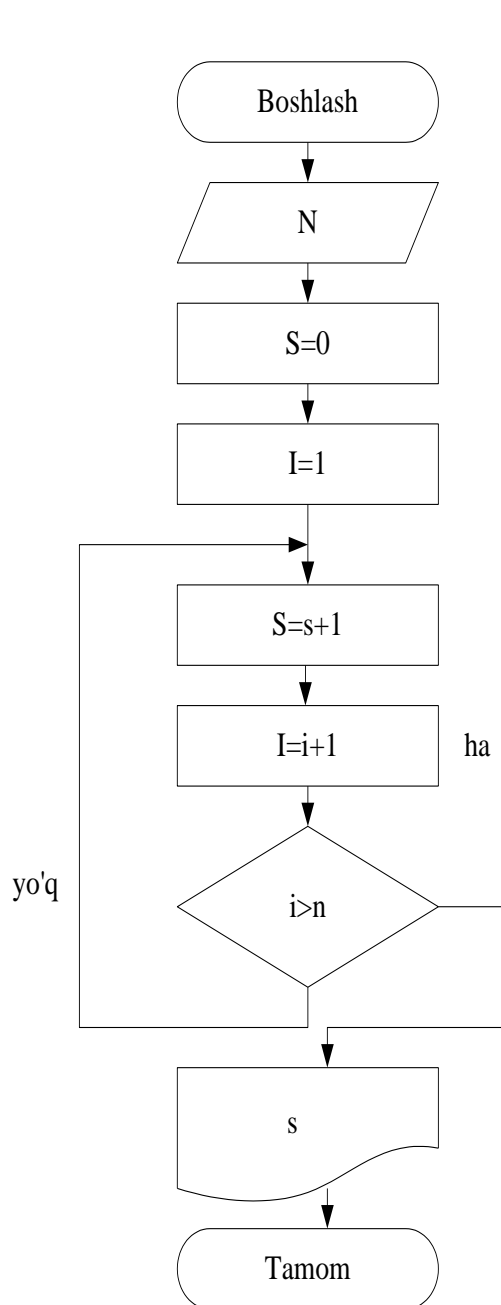
2-rasm. Tarmoqlanuvchi algoritmgaga doir misolning blok-sxemasi.

Takrorlanuvchi tuzilma.

Algoritmda ma'lum bir hisoblash jarayonlari ko'p marta takrorlanadi. Agar ma'lum bir harakatlar ketma-ketligi o'zgaruvchilarning turli qiymatlari uchun

takroran bajarilsa, bunday hisoblash jarayonlari takrorlanuvchi hisoblash jarayonlari deyiladi. Takrorlanuvchi hisoblash jarayonlarini ifodalash uchun takrorlanish tuzilmasidan foydalaniladi.

Misol: Berilgan 1 dan to N gacha bo'lgan natural sonlar yig'indisini hisoblash algoritmini blok sxema shaklida tuzing. N



$$S = \sum i$$

$$I = 1$$

1 Berilgan masalani 1 chi blokida sonlar soni (N) kiritiladi.

2 Ikkinchi blokda S yacheykasi 0 ga tenglashtiriladi.

3 Uchinchi blokda sikl tayyorlanadi, siklning parametrik I ga birinchi qiymat o'zlashtiriladi.

4 To'rtinchi blokda hisoblash amallari bajariladi.

5 Beshinchi blokda sikl parametrik birga oshiriladi.

6 Oltinchi, Shart o'tish blokda I parametri N bilan solishtiriladi va Shart bajarilishiga ko'ra boshqarish 4 chi yoki 7 chi blokka uzatiladi.

7 Yettinchi blokda hisoblash natijasida olingan S ning qiymati chiqariladi va keyingi blok bilan dastur o'z ishini tugallaydi.

3-rasm . Takrorlanuvchi algoritm blok-sxemasi.

Nazorat uchun savollar

1. Algoritm so'zi kim tomondan va qachon kiritilgan?
2. Algoritm so'zi nima ma'noni anglatadi?
3. Algoritmni qanday xossalari bilasiz?
4. Algoritmni qanday usullarda ifodalash mumkin?
5. Algoritmni so'z orqali ifodalasg deganda nimani tushunasiz?
6. Algoritmni grafik ifodalanishidagi bloklarni tushuntirib bering.
7. Algoritmni operatorli ifodalanishi nima?
8. Tarmoqlanuvchi jarayonlar deb nimaga aytiladi?
9. Tarmoqlanishda harakatlar qanday bajariladi?
10. Ayri tuzilmasi qanday? Uni izoxlang.
11. Takrorlanuvchi hisoblash jarayonlari deb nimaga aytiladi?
12. Takrorlanuvchi hisoblash jarayonlarini grafik ifodalaganda qanday grafik shakldan foydalaniladi?

BAJARISH UCHUN TOPSHIRIQLAR

Chiziqli algoritmga topshiriqlar.

Topshiriq tarkibi.

Berilgan arifmetik ifodalar qiymatlarini hisoblash dasturini tuzing. Ifoda va boshlang'ich ma'lumot qiymatlarini tanlash talabalarning jurnaldagi tartib raqamlariga mos holda topshiriq variantlari bilan aniqlanadi.

Topshiriq variantlari.

№	Ifoda	Qiymatlar
1	$a = 2^{-x} \sqrt{x + \sqrt[4]{ y }}, \quad b = \sqrt{e^{-x - \frac{1}{\sin(z)}}}$	<p>x = 3.981</p> <p>u = -1.625</p> <p>z = 0.512</p>
2	$a = y^{3\sqrt{ x }} + \cos^3(y-3), \quad b = \frac{y * \left(\arctg(z) - \frac{\pi}{6} \right)}{ x + \frac{1}{y^2 + 1}}$	<p>x = -6.251</p> <p>y = 0.824</p> <p>z = 25.001</p>

3	$a = 2^{(y^x)} + (3^x)^y, \quad b = \frac{ x-y * \left(1 + \frac{\sin^2(z)}{x+y}\right)}{e^{ x-y } + \frac{x}{2}},$	$x = 3.251$ $y = 0.325$ $z = 0.466$
4	$a = \frac{\sqrt{ x-1 } - \sqrt[3]{ y }}{1 + \frac{x^2}{x+2} + \frac{y^2}{y+4}}, \quad b = x * \left(\arctg(z) + e^{-(x+3)}\right),$	$x = -0.622$ $y = 3.325$ $z = 5.541$
5	$a = \sqrt[4]{y} + \sqrt[3]{x-1}, \quad b = x-y * (\sin^2(z) + \tg(z)),$	$x = 17.421$ $y = 10.365$ $z = 0.828$
6	$a = \frac{ y ^{x+1}}{\sqrt[3]{ x-2 } + 3} + \frac{x + \frac{y}{2}}{2 * x+y }, \quad b = (x+1) \frac{1}{\sin(z)},$	$x = 1.625$ $y = -15.400$ $z = 0.252$
7	$a = \frac{x^{y+1} + e^{y-1}}{1 + x * y - \tg(z) }, \quad b = 1 + \frac{ y-x }{ x } + \frac{ y-x ^2}{ y ^2} + \frac{ y-x ^3}{ z ^3},$	$x = 2.444$ $y = 0.869$ $z = -0.166$
8	$a = 1 + \frac{x}{\sin(x)} + \frac{x^2}{\sin^2(x)} + \frac{x^3}{\sin^3(x)} + \frac{x^4}{\sin^4(x)},$ $b = x * (\sin(\arctg(z) + \cos^2(y))),$	$x = 0.355$ $y = 0.025$ $z = 32.005$
9	$a = (1+y) * \frac{x + \frac{y}{x^2+4}}{y^{x-2} + \frac{1}{x^2+4}}, \quad b = \frac{1 + e^{y-2}}{\frac{x}{2} + \sin^2(z)},$	$x = 3.258$ $y = 4.005$ $z = -0.666$
10	$a = y + \frac{x}{y + \frac{x^2}{y + \frac{x^3}{y}}}, \quad b = (1 + \tg^2\left(\frac{z}{2}\right)) \sqrt{ y +6},$	$x = 0.1$ $y = -7.85$ $z = 0.765$
11	$a = \lg(\sqrt{e^{x-y}} + x y +z), \quad b = \frac{x}{\sin(x)} - \frac{x^3}{\sin^3(x)} + \frac{x^5}{\sin^5(x)},$	$x = 1.542$ $y = -3.261$ $z = 80.005$

12	$a = \frac{2 * \cos(x - \frac{\pi}{6})}{\frac{1}{7} + \sin^2 y}, \quad b = 1 + \frac{z^2 - 1}{3 + \frac{z^2 - 2}{5}},$	$x = 1.426$ $y = -1.220$ $z = 3.5$
13	$a = \frac{\sqrt{8 + x - y ^2 + 1.5 * x}}{x^2 + y^2 + 2}, \quad b = e^{ x - y } * \left(\operatorname{tg}^2(z) + 1 \right)^x,$	$x = -4.5$ $y = 0.75$ $z = 0.845$
14	$a = \frac{1 + \left(e^{x+y} + e^{x-y} \right)^2}{4 * \left x - \frac{2 * y}{1 + x^2 * y^2} \right } * x^{ y }, \quad b = \cos^2 \left(\operatorname{arctg} \left(\frac{1}{z} \right) \right),$	$x = 3.741$ $y = -0.825$ $z = 0.16$
15	$a = \frac{\left \cos(x) + \cos(y) \right ^{1 + 2 * \sin^2(y)}}{0.1 * (x - y)},$ $b = 1 + \frac{z}{\cos(y)} + \frac{z^2}{2 * \cos(y)} + \frac{z^3}{3 * \cos(y)} + \frac{z^4}{4 * \cos(y)},$	$x = 0.4$ $y = -0.875$ $z = -0.475$
16	$a = \sqrt{10 * (\sqrt[3]{x} + x^{y-2})}, \quad b = (\operatorname{arctg}(z))^2 + \frac{ x + y }{2 + y},$	$x = 16.55$ $y = -2.75$ $z = 0.15$
17	$a = \ln(y^{-\sqrt{ x }}) * \left(x - \frac{y}{2} \right), \quad b = \frac{2}{3} + \sin^2(\operatorname{arctg}(z)),$	$x = -15.246$ $y = 4.642$ $z = 20.001$
18	$a = \frac{5}{6} * \operatorname{arctg}(x) - \frac{3}{4} * \operatorname{arctg}(y), \quad b = \frac{x + 3 * x - y + x^2}{ x - y ^z + x^2},$	$x = -17.22$ $y = 6.33$ $z = 3.25$
19	$a = e^{ x - y } + \frac{ x - y ^{x + y}}{\sin^2(y)}, \quad b = \frac{\frac{1}{7} * \operatorname{arctg}(x) + \frac{6}{7} * \operatorname{arctg}(z)}{\sqrt[3]{z} - y},$	$x = -2.235$ $y = -0.823$ $z = 15.221$
20	$a = \left x^{\frac{y}{x}} - 3 \sqrt{\frac{y}{x + z}} \right , \quad b = (y - x) \frac{y - x}{1 + (y - x)^2},$	$x = 1.825$ $y = -18.225$ $z = -3.298$

21	$a = \frac{x + \frac{y}{5 + \sqrt{x}}}{ y - x + \sqrt{x}}, \quad b = e^{n-1} + \frac{\arctg(v)}{n+2},$	$x = 47.8$ $y = -5.5$ $n = -2.3$ $v = 0.8$
22	$a = y^x + \frac{\sqrt[3]{ x + y }}{\sin(n)}, \quad b = n + \frac{v^3}{n + \frac{v^2}{n + v^3}},$	$x = -0.85$ $y = 1.25$ $n = -0.22$ $v = 0.01$
23	$a = \frac{\sqrt[3]{x + 4\sqrt{ y }}}{x + \cos(5 + y)} * e^{-\frac{u}{x}}, \quad b = \sqrt{ y } * e^{-\left(y + \frac{u}{2}\right)},$	$x = 37.15$ $y = -12.55$ $u = 20.12$
24	$a = \frac{1}{2} * \left(x^{ y-x } + y^{\frac{(x+y)}{2}} \right), \quad b = \lg\left(\sqrt[3]{u} + \sqrt{v} + \frac{2}{3}\right),$	$x = 3.255$ $y = 2.981$ $u = 125.331$ $v = 33.075$
25	$a = \left(\frac{2}{3} + y^2\right) * \frac{x + \frac{y}{2}}{y^2 + \frac{1}{1 + y^2}}, \quad b = \sqrt{\sin^2(\arctg(u))} + \sqrt{ \cos(v) },$	$x = 0.22$ $y = -6.72$ $u = 10.05$ $v = 0.35$
26	$a = u \frac{x+y}{2} - 3 \frac{\sqrt{x-1}}{\sqrt{ y +1}}, \quad b = \sin^2\left(\frac{2}{5} \arctg(v)\right),$	$x = 12.65$ $y = -2.255$ $u = 3.205$ $v = 0.88$
27	$a = \arctg\left(\frac{\sqrt{3}}{3}\right) + \log \frac{1}{3} \left \frac{x+1}{x-1} \right * \cos\left(\pi * x + \frac{\pi}{3}\right),$ $b = \frac{e^y + e^{-y}}{1 + \sin^2\left(y + \frac{\pi}{4}\right)} * \sqrt[3]{1 + \frac{(y-1)^2}{27}},$	$x = 0.5$ $y = 0$

28	$a = \log_4 \left(x + \frac{1}{16} \right) * \sqrt[6]{x+64} * \left(e^{\frac{x}{2}} + e^{-\frac{x}{2}} \right),$ $b = \operatorname{arctg} \left(\frac{1}{\sqrt{3}} \right) * \left(\frac{(x-1)^2}{3} + \sqrt[5]{x+32} \right),$	x = 0
29	$a = \frac{e^{x^2+x} + 1}{\sqrt{x+16}} * \cos^2 \left(x + \frac{3}{4} * \pi \right), \quad b = \log_2 \left y - \frac{1}{8} \right * \sqrt[7]{2 * (y-8)^2},$	x = 0 y = 0
30	$a = \operatorname{arctg} \left(\frac{x}{\sqrt{3}} \right) * \lg \left \frac{x-101}{11-x} \right , \quad b = \sqrt[4]{(y-1)^2 + 81} * \left(1 + \sin^2 \left(\frac{\pi}{4} * y \right) \right),$	x = 1.0 y = 1.0

Tarmoqlanuvchi algoritmgga topshiriqlar.

№	Variantlar
1.	a,b,c uchta raqamdan eng katta kvadratni toping va qiymatni bosmaga chiqaring: N=1, agar – a eng katta bo’lsa; N=2, agar - b eng katta bo’lsa; N=3, agar - c eng katta bo’lsa;
2.	x_0, y_0 nuqta koordinatalari bilan birga r radiusli aylanaga tushadimi, aniqlang. $r^2=x^2+y^2$ aylananing tenglamasi. Agar nuqta aylananing ichida bo’lsa va N=0 aylanadan tashqari bo’lsa N=1 egallasin.
3.	x,u koordinatalar bilan nuqta qaysi kvadratda joylashganini aniqlang va bosmaga kvadratning raqamini chiqaring.
4.	a, b haqiqiy sonlar berilgan. Agar $a \geq b$ bo’lsa bu raqamlarni ikkiga ko’paytiring va agar shart bajarilmasa absolyut qiymatlari bilan almashtiring.
5.	a,b haqiqiy ikkita son berilgan “a, b dan katta” matnni va birinchi sonni chiqaring, agar u ikkinchisidan katta bo’lsa ikkala son va matn “a, b dan kichik”.
6.	Ikkita haqiqiy son berilgan. Birinchi sonni nol bilan almashtiring, agar u kichik bo’lsa yoki ikkinchisiga teng bo’lsa, aks holda uni o’zgartirmasdan qoldiring.

7.	Uchta haqiqiy son berilgan. (1;3) intervaliga tegishli sonlarni ulardan tanlang.
8.	x, y ($x \neq y$) haqiqiy sonlar berilgan. Bu ikkita sondan kichigini yarim miqdor bilan almashtiring, kattasini esa ikkiga ko'paytiring.
9.	Uchta haqiqiy son berilgan. Manfiy bo'lmagan qiymatlarni kvadratga o'tkazing.
10.	a, b va c haqiqiy sonlar berilgan. $a < b < c$ tengsizliklar bajarilayotganligini tekshiring. Agar shartlar bajarilayotgan bo'lsa " $a < b < c$ " bosmaga chiqaring. Aks holda «Shartlar bajarilmayapti» deb bosmaga chiqaring.
11.	a, b, c uchta haqiqiy sonlar berilgan. Ularni ikkiga ko'paytiring, agar $a \geq b \geq c$ bo'lsa, agar bunday bo'lmasa absolyut qiymatlar bilan almashtiring.
12.	Uchta haqiqiy sonlar berilgan. Ulardan manfiy qiymatlarni kvadratga chiqaring.
13.	x, y, z haqiqiy sonlar berilgan. (x, y, z) max toping.
14.	x, y, z haqiqiy sonlar berilgan. $\max (x+y+z, x \times y \times z)$ $\min ((x+y+z)/2, x \times y \times z) + 1$ hisoblang.
15.	Agar x, y, z juft hisobi har xil haqiqiy sonlarning miqdori 1 dan kichik bo'lsa, bunda bu uchta sondan eng kichigini ikkita boshqa yarimmiqdorli son bilan almashtiring, aks holda x va u yarimmiqdorli ikkita qolgan qiymatni almashtiring.
16.	x, y, z haqiqiy musbat sonlar berilgan. x, y, z . tomonlar uzunligi bilan uchburchak borligini aniqlang.
17.	a, b, c uchta harakatdagi sonlar berilgan. Ularni ikkiga ko'paytiring, agar $a \leq b \leq c$ bo'lsa, unday bo'lmasa ularni kvadrat bilan almashtiring.
18.	x, y ($x \neq y$) haqiqiy sonlar berilgan. Bulardan ikkita kichik sonni ularning summasi bilan almashtiring, kattasini esa uchga ko'paytiring.
19.	x, y, z haqiqiy sonlari uch juft hisobi undan katta bo'lsa shu uch sondan kattaroq sonni yarimsummaliga almashtiring yoki x va u kattalarini qolgan ikki

	yarimsummalining qiymatiga almashtiring.
20.	Uchta haqiqiy sonlar berilgan. Ulardan intervaliga tegishli bo'lmaganlarini toping (- 1;5).
21.	Aylananing S_1 va kvadratning S_2 yuzalari berilgan. Kvadrat aylanaga joylashdimi yo'qmi aniqlang. $S = \pi R^2$. Aylananing maydoni.
22.	Aylananing S_1 va kvadrata S_2 maydoni berilgan. Aylana kvadrat ichiga sig'adimi, yo'qmi aniqlang.
23.	(x, u,z) koordinatalari bilan nuqta berilgan. Nuqta $\frac{x^2}{9} + \frac{y^2}{4} + \frac{z^2}{1} = 1$, ellipsoidni ichida joylashgan, yoki yo'qmi aniqlang. Agar joylashgan bo'lsa bosmaga «tegishli», agar joylashmagan bo'lsa «tegishli emas» deb bosmaga chiqaring.
24.	(x, u,z) koordinatalari bilan nuqta berilgan. Nuqta $\frac{x^2}{25} + \frac{y^2}{9} = 1$, ellipsning ichida joylashgan, yoki yo'qmi aniqlang. Agar joylashgan bo'lsa bosmaga «tegishli», agar joylashmagan bo'lsa «tegishli emas» deb bosmaga chiqaring.
25.	(x, u,z) koordinatali nuqta berilgan. $(x-1)^2 + y^2 + (z+2)^2 = 16$ nuqta shu sferani ichida joylashganligini tekshiring, agar joylashgan bo'lsa bosmaga chiqaring.
26.	x,y,z haqiqiy sonlar berilgan. (x,y,z) min toping.
27.	A haqiqiy son berilgan. Agar f(x) bo'lsa, hisoblang va bosmaga chiqaring. $F(x) = \begin{cases} 0, & \text{àgar } x \leq 0 \\ x^2 < x, & \text{àgar } 0 < x \leq 1 \\ x^2 < \sin \pi x^2, & \text{boshqa hollarda} \end{cases}$
28.	Uchta haqiqiy son berilgan. f(a) ni tanglang va bosmaga chiqaring agar $F(x) = \begin{cases} x^2, & \text{àgar } -z \leq x < z \\ 4, & \text{boshqa hollarda} \end{cases}$
29.	Haqiqiy son berilgan. Hisoblang va bosmaga chiqaring, agar $F(x) = \begin{cases} x^2 + 4x + 5, & \text{àgar } x \leq 2 \\ 1/(x^2 + 4x + 5), & \text{boshqa hollarda} \end{cases}$

Takrorlanuvchi algoritmga topshiriqlar.

№	Miqdor	a, b diapazoni	y-funksiyasi	N
1.	$s = 1 + \frac{\ln 3}{1!}x + \frac{\ln^2 3}{2!}x^2 + \dots + \frac{\ln^3 3}{n!}x^n$	$0,1 \leq x \leq 1$	$y = 3^x$	10
2.	$s = \cos x + \frac{\cos 2x}{2} + \dots + \frac{\cos nx}{n}$	$\frac{\pi}{5} \leq x \leq \frac{9\pi}{5}$	$y = -\ln \left 2 \sin \frac{x}{2} \right $	40
3.	$s = x - \frac{x^3}{3!} + \dots + (-1)^n \frac{x^{2n+1}}{(2n+1)!}$	$0,1 \leq x \leq 1$	$y = \sin x$	40
4.	$s = \sin x \frac{\sin 2x}{2} + \dots + (-1)^{n+1} \frac{\sin x}{n}$	$\frac{\pi}{5} \leq x \leq \frac{4\pi}{5}$	$y = \frac{\pi}{2}$	10
5.	$s = 1 + \frac{x}{1!} + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$	$1 \leq x \leq 2$	$y = e^x$	40
6.	$s = 1 + \frac{\cos \frac{\pi}{4}}{1!}x + \dots + \frac{\cos n \frac{\pi}{4}}{n!}x^n$	$0,1 \leq x \leq 1$	$y = e^{x \cos \frac{\pi}{4}} \cos \left(\sin \frac{\pi}{4} \right)$	25
7.	$s = 1 - \frac{x^2}{2!} + \dots + (-1)^n \frac{x^{2n}}{(2n)!}$	$0,1 \leq x \leq 1$	$y = \cos x$	10
8.	$s = x \sin \frac{\pi}{4} + x^2 \sin 2 \frac{\pi}{4} + \dots + x^n \sin \frac{\pi}{4}$	$0,1 \leq x \leq 0,8$	$y = \frac{x \sin \frac{\pi}{4}}{1 - 2x \cos \frac{\pi}{4} + x^2}$	20
9.	$s = x + \frac{x^5}{5} + \dots + \frac{x^{4n+1}}{4n+1}$	$0,1 \leq x \leq 0,8$	$y = \frac{1}{4} \ln \frac{1+x}{1-x} + \frac{1}{2} \operatorname{arctg} x$	30
10.	$s = 1 + \frac{\cos x}{1!} + \dots + \frac{\cos nx}{n!}$	$0,1 \leq x \leq 1$	$y = e^{\cos x} * \cos(\sin x)$	25
11.	$s = 1 + 3x^2 + \dots + \frac{2n+1}{n!} x^{2n} s$	$0,1 \leq x \leq 1$	$y = (1 + 2x^2) e^{x^2}$	10
12.	$s = \frac{x \cos \frac{\pi}{3}}{1} + \frac{x^2 \cos 2 \frac{\pi}{3}}{2} + \dots + \frac{x^n \cos n \frac{\pi}{3}}{n}$	$0,1 \leq x \leq 0,8$	$y = -\frac{1}{2} \ln \left(1 - 2x \cos \frac{\pi}{3} \right)$	35
13.	$s = \frac{x-1}{x+1} + \frac{1}{3} \left(\frac{x-1}{x+1} \right)^3 + \frac{1}{2n+1} \frac{(x-1)}{(x+1)^{2n+1}}$	$0,2 \leq x \leq 1$	$y = \frac{1}{2} \ln x$	10
14.	$s = -\cos x + \frac{\cos 2x}{2^2} + \dots + (-1)^n \frac{\cos nx}{n^2}$	$\frac{\pi}{4} \leq x \leq \pi$	$y = \left(x^2 - \frac{\pi^2}{4} \right)$	20
15.	$s = \frac{x^3}{3} - \frac{x^5}{15} + \dots + (-1)^{n+1} \frac{x^{2n+1}}{4n^2 - 1}$	$0,1 \leq x \leq 1$	$y = \frac{1+x^2}{2} \operatorname{arctg} x - \frac{x}{2}$	30
16.	$s = \sin x + \frac{\sin 3x}{3x} + \dots + \frac{\sin(2n-1)x}{2n-1}$	$\frac{\pi}{10} \leq x \leq \frac{9\pi}{10}$	$y = \frac{\pi}{4x}$	20
17.	$s = 1 + \frac{x^2}{2!} + \dots + \frac{x^{2n}}{(2n)!}$	$0,1 \leq x \leq 1$	$y = \frac{e^x + e^{-x}}{2}$	20
18.	$s = \frac{\cos 2x}{3} + \frac{\cos 4x}{15} + \dots + \frac{\cos 2nx}{4n^2 - 1}$	$0,1 \leq x \leq 0,8$	$y = \frac{1}{2} - \frac{\pi}{4} \sin x $	50

19.	$s = 1 + \frac{2x}{!} + \dots + \frac{(2x)^n}{n!}$	$0,1 \leq x \leq 1$	$y = e^{\frac{x}{2}}$	20
20.	$s = 1 + 2\frac{x}{2} + \dots + \frac{n^2 + 1}{n!} \left(\frac{x}{20}\right)^n$	$0,1 \leq x \leq 1$	$y = \left(\frac{x^2}{4} + \frac{x}{2} + \frac{1}{2}\right) e^{\frac{x}{2}}$	30
21.	$s = x - \frac{x^3}{3} + \dots + (-1)^n \frac{x^{2n+1}}{2n+1}$	$0,1 \leq x \leq 0,5$	$y = \arctg x$	40
22.	$s = 1 - \frac{3}{2}x^2 + \dots + (-1)^n \frac{2n^2 + 1}{(2n)!} x^{2n}$	$0,1 \leq x \leq 1$	$y = \left(1 - \frac{x^2}{x}\right) \cos x - \frac{x \sin x}{2}$	35
23.	$s = -\frac{(2x)^2}{2} + \frac{(2x)^2}{24} + \dots + (-1)^n \frac{(2x)^{2n}}{(2n)!}$	$0,1 \leq x \leq 1$	$y = 2(\cos^2 x - 1)$	15
24.	$s = -(1+x)^2 + \frac{(1+x)^4}{2} + \dots + (-1)^n \frac{(1+x)^2}{n}$	$-2 \leq x \leq -0,1$	$y = \ln \frac{1}{2+2x+x^2}$	40
25.	$s = x + \frac{x^3}{3!} + \dots + \frac{x^{2n+1}}{(2n+1)!}$	$0,1 \leq x \leq 1$	$y = \frac{e^x - e^{-x}}{2}$	20
26.	$s = \frac{x}{3!} + \frac{4x^2}{5!} + \dots + \frac{n^2}{(2n+1)} x^n$	$0,2 \leq x \leq 0,8$	$y = \frac{1}{4} \left(\frac{x+1}{\sqrt{x}}\right) sh\sqrt{x} - ch\sqrt{x}$	20
27.	$s = x \cos \frac{\pi}{4} + x^2 2 \frac{\pi}{4} + \dots + x^n \cos n \frac{\pi}{4}$	$0,1 \leq x \leq 0,8$	$y = \frac{x \cos x \frac{\pi}{4} - x^2}{1 - 2x \cos \frac{\pi}{4} + x^2}$	40
28.	$s = 3x + 8x^2 + \dots + n(n+2)x^n$	$0,1 \leq x \leq \pi$	$y = \frac{x(3-x)}{(1-x)^3}$	40
29.	$s = \cos x + \frac{\cos 3x}{32} + \dots + \frac{\cos(2n-1)x}{(2n-1)^2}$	$\frac{\pi}{5} \leq x \leq \pi$	$y = \frac{\pi^2}{8} - \frac{\pi}{4} x $	40
30.	$s = \frac{x^2}{2} - \frac{x^4}{12} + \dots + (-1)^{n+1} \frac{x^{2n}}{2n(2n-1)}$	$0,1 \leq x \leq 0,8$	$y = x \arctg(x)$	20

4-Laboratoriya ishi. Chiziqli algoritm dasturlari

Ishning maqsadi

1. C++ tilidagi dastur tarkibi.
2. Preprotessor direktivalari.
3. Oqim bilan kiritish va chiqarish.
4. Formatli kiritish va chiqarish.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarning dasturini tuzishi lozim.
2. Hisobot shaklida oldin vazifa, uni bajarishda foydalanilgan standart funksiyalar, dastur kodi va bajarishdan hosil bo'lgan natijani keltirish lozim.
3. 4-laboratoriya ishida beriladigan barcha vazifalarni bitta hisobot shaklida topshiring.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. Laboratoriya ishidagi topshiriq raqami.
3. Topshiriq kodi.
4. Topshiriq natijasi
5. Dasturning elektron versiyasi (Albatta bu dasturlar kompyuterda o'qituvchiga ko'rsatiladi. O'qituvchini talab va takliflariga ko'ra dastur to'g'rilanadi va so'ngra hisobot tayyorlanadi.).

NAZARIY QISM

CHIZIQLI DASTURLAR

Alifbo. C tili alifbosiga quyidagi simvollar kiradi.

- Katta va kichik lotin alifbosi harflari (A, B, ..., Z, a, b, ..., z)
- Raqamlar: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

- Maxsus simvollar: “, { } | [] () + - / % \ ; ‘ . : ? < = > _ ! & * # ~ ^
- Ko’rinmaydigan simvollar ("umumlashgan bo’shliq simvollari").

Leksemalarni o’zaro ajratish uchun ishlatiladigan simvollar (misol uchun bo’shliq, tabulyatsiya, yangi qatorga o’tish belgilari).

Izohlarda, satrlarda va simvolli konstantalarda boshqa literallar, masalan rus harflari ishlatilishi mumkin.

C++ tilida olti xil turdagi leksemalar ishlatiladi: erkin tanlanadigan va ishlatiladigan identifikatorlar, xizmatchi so’zlar, konstantalar (konstanta satrlar), amallar (amallar belgilari) va ajratuvchi belgilar.

Identifikator. Identifikatorlar lotin harflari, ostki chiziq belgisi va sonlar ketma-ketligidan iborat bo’ladi. Identifikator lotin harfidan yoki ostki chiziq belgisidan boshlanishi lozim.

Misol uchun:

A1, _MAX, adress_01, RIM, rim

Katta va kichik harflar farqlanadi, shuning uchun oxirgi ikki identifikator bir-biridan farq qiladi.

Borland kompilyatorlaridan foydalanilganda nomning birinchi 32 harfi, ba’zi kompilyatorlarda 8 ta harfni inobatga oladi. 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 so’zlar mavjud:

Turlar nomlari: char, short, int, unsigned, long, float, double.

Operatorlar nomlari: if, else, switch, case, while, do, for, default, break, continue, goto.

Xotira turlari: auto, register, static, extern.

Turlar bilan ishlash: typedef, sizeof.

Tuzilma: struct, union.

Chiqish: return, entry.

O'zgaruvchilarni ta'riflash. C++ tilida o'zgaruvchini aniqlash uchun kompyuterga uning turi (masalan, int, char yoki float) hamda ismi haqida ma'lumot beriladi. Bu axborot asosida kompilyatorga o'zgaruvchi uchun qancha joy ajratish lozim va bu o'zgaruvchida qanday turdagi qiymat saqlanishi mumkinligi haqida ma'lumot aniq bo'ladi. O'zgaruvchi nomi identifikator bo'lib, xizmatchi so'zlardan farqli bo'lishi kerak.

Har bir yacheyka bir bayt o'lchovga ega. Agar o'zgaruvchi uchun ko'rsatilgan tur 4 baytni talab qilsa, uning uchun to'rtta yacheyka ajratiladi. Aynan o'zgaruvchini turiga muvofiq ravishda kompilyator bu o'zgaruvchi uchun qancha joy ajratish kerakligini aniqlaydi.

Kompyuterda qiymatlarni ifodalash uchun bitlar va baytlar qo'llaniladi va xotira baytlarda hisoblanadi.

O'zgaruvchilar turlari. O'zgaruvchilarning quyidagi turlari mavjud:

char – bitta simvol;

long char – uzun simvol;

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 ko'rilgan turlar oldiga unsigned (ishorasiz) ta'rifi qo'shilishi mumkin. Bu ta'rif qo'shilgan butun sonlar ustida amallar mod 2^n arifmetikasiga asoslangandir. Bu yerda n soni int turi xotirada egallovchi razryadlar sonidir. Agar ishorasiz k soni uzunligi int soni razryadlar sonidan uzun bo'lsa, bu son qiymati $k \bmod 2^n$ ga teng bo'ladi. Ishorasiz k son uchun – k amali $2^n - k$ formula asosida hisoblanadi. Ishorali, ya'ni signed turidagi sonlarning eng katta razryadi son ishorasini ko'rsatish uchun ishlatilsa unsigned (ishorasiz) turdagi 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;

Konstantalar turlari. Konstanta bu o'zgartirish mumkin bo'lmagan qiymatdir. C tilida besh turdagi konstantalar ishlatilishi mumkin: simvollar, butun sonlar, haqiqiy sonlar, sanovchi konstantalar va nol ko'rsatkich.

Belgili o'zgarmaslar. Belgili o'zgarmaslar odatda bir bayt joyni egallaydi va bu 256 xil belgini saqlash uchun yetarlidir. Char turi qiymatlarini 0..255 sonlar to'plamiga yoki ASCII belgilar to'plamiga interpretatsiya qilish mumkin.

ASCII belgilari deganda kompyuterlarda qo'llaniladigan standart belgilar to'plami tushuniladi. ASCII - bu American Standard Code for Information Interchange (Amerikaning axborot almashinishi uchun standart kodi) degan ma'noni anglatadi.

Misol uchun, 'x','*','\012','\0','\n'- bitta simvolli konstanta; 'dd','\n\t','\x07\x07' ikki simvolli konstantalar.

C++ kompilyatorida matnlarni formatlovchi bir nechta maxsus belgilardan foydalaniladi (Ulardan eng ko'p tarqalgani jadvalda keltirilgan).

Maxsus belgilar axborotlarni ekranga, faylga va boshqa chiqarish qurilmalariga chiqarishda formatlash uchun qo'llaniladi.

Maxsus '\ ' simvolidan boshlangan simvollar eskeyp simvollar deyiladi. Simvolli konstanta qiymati simvolning kompyuterda qabul qilingan sonli kodiga tengdir.

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'shish va - ayirish amallariga, hamda multiplikativ, ya'ni * ko'paytirish, / bo'lish va % modul olish amallariga ajratiladi.

Butun sonni butun songa bo'lganda natija butun songacha yaxlitlanadi. 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.

Unar amallarga ishorani o'zgartiruvchi unar minus – va unar plyus + amallari kiradi. Bundan tashqari inkrement ++ va dekrement -- amallari ham unar amallarga kiradi.

Inkrement ++ 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 esa i++ ko'rinishda ishlatish oldin o'zgaruvchi qiymatidan foydalanib, so'ngra oshirish kerakligini ko'rsatadi. Misol uchun, i ning 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 ning qiymati 3 ga teng bo'ladi.

Dekrement -- 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 mumkin.

Munosabat amallari. Munosabat amallari qiymatlari 1 ga teng agar munosabat bajarilsa va aksincha 0 ga tengdir. Munosabat amallari arifmetik turdagi operandlarga yoki ko'rsatkichlarga qo'llaniladi.

Misollar:

$1! = 0$ qiymati 1 ga teng;

$1 == 0$ qiymati 0 ga teng;

$3 > = 3$ qiymati 1 ga teng;

$3 > 3$ qiymati 0 ga teng;

$2 < = 2$ qiymati 1 ga teng;

$2 < 2$ qiymati 0 ga teng;

Katta >, kichik <, katta yoki teng >= , kichik yoki teng <= amallarining ustuvorligi bir xildir.

Teng == va teng emas != amallarining ustuvorligi o'zaro teng va qolgan amallardan pastdir.

Mantiqiy amallar. C++ tilida mantiqiy tur yo'q. Shuning uchun mantiqiy amallar 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 ustuvorligi oshib borish tartibida berilgandir.

Inkor ! amali unar qolganlari binar amallardir.

Qiymat berish amali. Qiymat berish amali = binar amal bo'lib chap operandi odatda o'zgaruvchi o'ng operandi esa ifodaga teng bo'ladi. Misol uchun

$z = 4.7+3.34$

Bu qiymati 8.04 ga teng ifodadir. Bu qiymat z- o'zgaruvchiga ham beriladi.

Bu ifoda oxiriga nuqta vergul (;) belgisi qo'yilganda operatorga aylanadi.

$z = 4.7+3.34$

Bitta ifodada bir necha qiymat berish amallari qo'llanilishi mumkin. Misol uchun:

$c = y = f = 4.2+2.8;$

Bundan tashqari C tilida murakkab qiymat berish amali mavjud bo'lib, umumiy ko'rinishi quyidagichadir:

O'zgaruvchi_nomi **amal** = ifoda;

Bu yerda **amal** quyidagi amallardan biri *,/,%,+,-, &,^,|, <<, >>.

Misol uchun:

$x+ = 4$ ifoda $x = x+4$ ifodaga ekvivalentdir;

$x* = a$ ifoda $x = x*a$ ifodaga ekvivalentdir;

$x/ = a+b$ ifoda $x = x/(a+b)$ ifodaga ekvivalentdir;

$x>> = 4$ ifoda $x = x>>4$ ifodaga ekvivalentdir;

C tilida dastur tuzilishi

Sodda dastur tuzilishi. Dastur preprotssessor komandalari va bir necha funksiyalardan iborat bo'lishi mumkin. Bu funksiyalar orasida **main** nomli asosiy funksiya bo'lishi shart. Agar asosiy funksiya boshqa funksiyalar ishlatilmasa dastur quyidagi ko'rinishda tuziladi:

```
Preprotssessor_komandalari
void main()
{
Dastur tanasi
}
```

Preprotssessor direktivalari kompilyasiya jarayonidan oldin preprotssessor tomonidan bajariladi. Natijada dastur matni preprotssessor direktivalari asosida o'zgartiriladi.

Preprotssessor komandalaridan ikkitasini ko'rib chiqamiz.

#include <fayl_nomi> Bu direktiva standart bibliotekalardagi funksiyalarni dasturga joylash uchun foydalaniladi.

#define <almashtiruvchi ifoda> <almashinuvchi ifoda>

Bu direktiva bajarilganda dastur matnidagi almashtiruvchi ifodalar almashinuvchi ifodalarga almashtiriladi.

Misol tariqasida C tilida tuzilgan birinchi dasturni keltiramiz:

```
#include <stdio.h>
void main()
{
printf("\n Salom, Dunyo! \n");
}
```

Bu dastur ekranga **Salom, Dunyo!** jumlasini chiqaradi.

Almashtiruvchi define direktivasi yordamida bu dasturni quyidagicha yozish mumkin:

```
#include <stdio.h>
```

```

#define begin {
#define yend }
#define pr printf("\n Salom, Dunyo! \n");
void main()
begin
pr;
end

```

Ma'lumotlarni kiritish va chiqarish

Ma'lumotlarni kiritish/chiqarish (Input / Output), ular ustida amallarni bajarish va natijalarni olish C++ da oqim obektlari orqali bajarilishi mumkin. Kiritish/chiqarishni C dagi kabi funksiyalar bilan ham amalga oshirsa bo'ladi. C++ falsafasiga ko'ra har bir kiritish/chiqarish jihozi (ekran, printer, klaviatura va h.k.) baytlar oqimi bilan ishlagandek qabul qilinadi. Normal holatda bu oqimlar ekranga va klaviaturaga ulangan bo'ladi. Bu oqim ikki xil : >> va << ko'rinishda qabul qilingan. Ma'lumotlarni chiqarish oqimi cout << ko'rinishida va ma'lumotlarni kiritish cin >> ko'rinishida yoziladi. cout (console output) va cin (console input) ma'nolarini bildiradi.

Ma'lumotlarni kiritishda cin so'zidan keyin o'zgaruvchilarning ismlari keltiriladi. Agar ular bir nechta bo'lsa, har bir o'zgaruvchi orasiga >> belgisi qo'yiladi.

Masalan: cin >> a >> b; cin >>GG >>lola >> a12; ,bu o'zgaruvchilarning son qiymatlari dastur kompilyatsiyadan o'tganidan keyin klaviatura orqali beriladi. Sonlarni bo'sh joy orqali yoki enter klavishi orqali kiritish mumkin. cin operatori orqali faqat son qiymatlar kiritiladi, ifodalar yozilishi mumkin emas.

Ma'lumotlarni chiqarish uchun cout << operatori ishlatiladi. Bu yerda o'zgaruvchi nomlari, qo'shtirnoq ichida istalgan so'z yoki gaplar yozilishi mumkin. Agar natijasi berilayotgan o'zgaruvchilar bir nechta bo'lsa ularni alohida holatda yoki ketma-ket chiqarilishi mumkin. Masalan:

```
cout << a; cout << a << b;
cout << "y=" << y;
cout << "funksiyaning qiymati teng =" << f;
```

Chiqarish oqimini kiritish oqimi bilan birga qo'llansa, dastur ko'rinish tushunarli va chiroyliroq bo'ladi. Masalan:

```
cout << "1 - sonni kiriting";
cin >> x;
cout << "2 - sonni kiriting";
cin >> y;
```

Bir nechta natijalar chiqarilayotgan bo'lsa, ular ekranda ketma-ket ko'rinishda namoyon bo'ladilar. Bu holat esa natijalarni o'qishda noqulaylik tug'dirishi mumkin. Shuning uchun `cout <<` oqimi oxirida `endl` (end line-satr oxiri) so'zi qo'yilsa, kursor keyingi qatorga o'tadi va keyingi natija ko'rinadi. Masalan: `cout << "Natijalar:" << endl;`

```
cout << "1-funksiyaning qiymati=" << f1 << endl;
cout << "2-funksiyaning qiymati=" << f2 << endl;
```

Ekranda quyidagi javoblar ko'rinadi:

Natijalar:

1-funksiyaning qiymati=2.35847

2-funksiyaning qiymati=-95.3687

`cout <<` ichida bevosita arifmetik amallarni ham keltirsa bo'ladi.

```
cout << "a*sin(b) ning javobi=" << a*sin(b) << endl;    yoki
```

```
cout << "a*sin(b) ning javobi=" << endl;
```

```
cout << a*sin(b) << endl;
```

Formatli chiqarish – printf. Chiqarish `printf` funksiyasi ko'rsatilgan parametrlarni standart oqimga chiqarish uchun ishlatiladi. Standart oqim tushunchasi keyingi boblarda yoritiladi. Hozircha standart oqim sifatida monitor tushunilishi yetarlidir.

Funksiya **stdio.h** modulida joylashgan bo'lib, umumiy ko'rinishi quyidagichadir:

printf(control,arg1,arg2,...)

Bunda control boshqaruvchi qator deb atalib ikki turdagi simvollardan iborat bo'ladi: oddiy chiqariluvchi simvollar va navbatdagi parametrni o'zgartirib chiqaruvchi spesifikasiyalar.

Har bir spesifikasiya % simvolidan boshlanib o'zgartirish turini ko'rsatuvchi simvol bilan tugaydi.

O'zgartirish simvollari quyidagilardan iborat.

Butun sonlar uchun:

d – parametr ishorali o'nlik butun songa aylantiriladi.

u – parametr ishorasiz o'nlik butun songa aylantiriladi.

0 – parametr ishorasiz va birinchi raqami 0 bo'lmagan sakkizlik songa aylantiriladi.

x – parametr ishorasiz va 0x belgisiz o'n oltilik songa aylantiriladi.

X – parametr xuddi "x" kabi. Faqat harf bilan ko'rsatiluvchi raqamlar katta harf ya'ni A,B,C,D,E,F sifatida yoziladi.

Haqiqiy sonlar uchun:

e – parametr float yoki double turidagi son deb qaraladi va ishorali m.nnnnnne+-xx ko'rinishidagi o'nlik songa keltiriladi.

E – parametr xuddi e kabi. Faqat mantissa belgisi katta harf ya'ni E sifatida yoziladi.

f – parametr float yoki double turidagi son deb qaraladi va ishorali m.nnnnnn ko'rinishidagi o'nlik songa keltiriladi.

g – parametr berilgan son qiymati va aniqligi uchun eng ixcham %ye yoki %f tanlaydi.

G – parametr xuddi g kabi. Faqat mantissa belgisi katta harf ya'ni G sifatida yoziladi.

Simvol va satr uchun:

c – parametr bitta simvol deb qaraladi.

s – parametr satr simvollar nolinchisi simvol uchramaguncha yoki ko'rsatilgan sondagi simvollar bosiladi.

Misol:

```
#include <stdio.h>

int main()
{
int num = -27; int number = 27; float f = 123.456;
char r = 'a'; char str[4] = "abc";
printf("%d\n", num); /* -27 */
printf("%u\n", number); /* 27 */
printf("%o\n", number); /* 33 */
printf("%x\n", number); /* lb */
printf("%f\n", f); /* 123.456001 */
printf("%ye\n", f); /* 1.23456ye+02 */
printf("%Ye\n", f); /* 1.23456Ye+02 */
printf("%c\n", r); /* a */
printf("%s\n", str); /* abc */
return 0;
}
```

Prosent % belgisi va o'zgartirish simvoli orasiga quyidagi simvollarni qo'yish mumkin.

Formatli kiritish Scanf. Scanf funksiyasi stdio.h modulida joylashgan bo'lib, umumiy ko'rinishi quyidagichadir:

Scanf(control, arg1, arg2,...)

Funksiya standart oqimdan simvollarni o'qib boshqaruvchi qator asosida formatlab mos parametrlarga yozib qo'yadi. Parametr ko'rsatkich bo'lishi lozim.

Uchburchak masalasi. Uchburchakning 2ta tomoni va ular orasidagi burchagi berilgan. Uchburchakning 3-tomonini, uning yuzasini, unga tashqi va ichki chizilgan aylana radiuslarini va qolgan burchaklarini topish dasturini tu'zing.

Demak: a, b, γ berilgan. Topish kerak: c, s, R, r - ?

$$c = \sqrt{a^2 + b^2 - 2ab \cos \gamma}; \quad s = \frac{ab \sin \gamma}{2};$$

$$r = \frac{abc}{4s}; \quad R = \frac{2s}{a+b+c}; \quad \frac{a}{\sin \alpha} = \frac{b}{\sin \beta} = \frac{c}{\sin \gamma} = 2R$$

```
# include <iostream.h>
# include <math.h>
void main ( )
{ float a, b, c, r, R, s, alf, bet, gam, x, pi=3.14;
cout << "uchburchak tomonlarini kiriting:\n";
cin >>a>>b;
  gam=pi/3; c = sqrt (a*a+b*b-2*a*b*cos(gam));
s = a*b*sin(gam)/2;
r = a*b*c / (4*s);
R = 2*s / (a+b+c);
x = b *sin(gam) / c;
bet = atan (sqrt (1- x*x)/x);
alf = pi – bet – gam;
cout << "Natijalar:"<<endl;
cout << "c="<<c<<endl;
cout << "r="<<r<<endl;
cout << "R="<<R<<endl;
cout << "s="<<s<<endl;
cout << "alfa="<<alf<<endl;
cout << "betta="<<bet<<endl;
}
```

Eslatma:

$$\arcsin x = \operatorname{arctg} \frac{x}{\sqrt{1-x^2}}$$

$$\arccos x = \operatorname{arctg} \frac{\sqrt{1-x^2}}{x}$$

Nazorat savollari

1. Kiritish va chiqarishni oqim bilan amalga oshirish?
2. Formatli kiritish va chiqarish.
3. O'zgaruvchi va konstantalarni tavsiflash.
4. Dasturlarni rasmiylashtirish?

BAJARISH UCHUN TOPSHIRIQLAR

Uchburchakning berilgan uchta parametriga asoslanib, qolgan yetita parametrini aniqlang.

№	Chiqish ma'lumotlari	Hisoblanayotgan o'sib borishlar
1.	$a = 3; b = 4; c = 5$	$P, S, R, r, \alpha, \beta, \gamma$
2.	$a = 4; b = 4; \gamma = \frac{\pi}{3}$	$c, P, S, R, r, \alpha, \beta$
3.	$b = 5; \alpha = \frac{\pi}{4}; \beta = \frac{\pi}{2}$	a, c, P, S, R, r, γ
4.	$b = 4; \alpha = \frac{\pi}{6}; \gamma = \frac{\pi}{2}$	a, c, P, S, R, r, β
5.	$a = 5; b = 4; \gamma = \frac{\pi}{6}$	$c, P, S, R, r, \alpha, \beta$
6.	$a = 3; b = 4; P = 12,5$	$c, S, R, r, \alpha, \beta, \gamma$
7.	$a = 2; R = \sqrt{2}; \gamma = \frac{\pi}{4}$	$b, c, P, S, r, \alpha, \beta$
8.	$a = 3; b = 4; \alpha = \frac{\pi}{4}$	$c, P, S, R, r, \beta, \gamma$
9.	$a = 4; c = 3; \beta = \frac{\pi}{2}$	$b, P, S, R, r, \alpha, \gamma$
10.	$a = 2; b = 4; S = 2\sqrt{3}$	$c, P, R, r, \alpha, \beta, \gamma$
11.	$a = 2; b = 2; \gamma = \frac{\pi}{4}$	$c, P, S, R, r, \alpha, \beta$
12.	$c = 6; \alpha = \frac{\pi}{4}; \beta = \frac{\pi}{5}$	a, b, P, S, R, r, γ

13.	$R = 2,5; \alpha = \frac{\pi}{6}; \beta = \frac{\pi}{3}$	a, b, c, P, S, r, γ
14.	$a = 5; b = 3; P = 12$	$c, R, S, r, \alpha, \beta, \gamma$
15.	$S = 12; \alpha = \frac{\pi}{6}; \beta = \frac{\pi}{3}$	a, b, c, P, R, r, γ
16.	$b = 5; c = 6; S = 3,125$	$a, P, R, r, \alpha, \beta, \gamma$
17.	$a = 10; b = 8; c = 6$	$P, R, S, r, \alpha, \beta, \gamma$
18.	$a = b = 3; \gamma = \frac{\pi}{4}$	$c, P, R, S, r, \alpha, \beta$
19.	$b = 15; \alpha = \frac{\pi}{3}; \beta = \frac{\pi}{4}$	a, c, P, R, S, r, γ
20.	$R = 2\sqrt{2}; \alpha = \beta = \frac{\pi}{4}$	a, b, c, P, S, r, γ
21.	$b = 8; \alpha = \frac{\pi}{5}; \gamma = \frac{\pi}{6}$	a, c, P, R, S, r, β
22.	$a = b = 2; \alpha = \frac{\pi}{4}$	$c, P, R, S, r, \beta, \gamma$
23.	$a = 4; R = 2\sqrt{2}; \gamma = \frac{\pi}{4}$	$b, c, P, S, r, \beta, \gamma$
24.	$a = 6; b = 4; \alpha = \frac{\pi}{4}$	$c, P, R, S, r, \beta, \gamma$
25.	$a = 6; b = 8; P = 5$	$c, R, S, r, \alpha, \beta, \gamma$
26.	$b = 5; c = 6; \alpha = \frac{\pi}{3}$	$a, P, R, r, S, \beta, \gamma$
27.	$a = 3; c = 5; \beta = \frac{\pi}{6}$	$b, P, S, r, R, \alpha, \gamma$
28.	$a = 4; R = 2\sqrt{2}; \beta = \frac{\pi}{4}$	$b, c, P, S, r, \alpha, \gamma$

29.	$R = 2,5; \alpha = \frac{\pi}{3}; \beta = \frac{\pi}{6}$	a, b, c, P, S, r, γ
30.	$b = 5; c = 6; S = 3,125$	$a, P, R, r, \alpha, \beta, \gamma$

5-Laboratoriya ishi. Tarmoqlanuvchi jarayonlarni dasturlash.

Ishning maqsadi

1. Shartli operatorning qisqa ko'rinishini qo'llash.
2. Shartli operatorning uzun ko'rinishini qo'llash.
3. Tanlov asosida tarmoqlanishni tatbiq eta olish.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarning dasturini tuzishi lozim.
2. Hisobot shaklida oldin vazifa, uni bajarishda foydalanilgan funksiyalar, dastur kodi va bajarishdan hosil bo'lgan natijalarni keltirishi lozim.
3. 5-laboratoriya beriladigan barcha vazifalarni bitta hisobot shaklida topshiring.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. Laboratoriya ishidagi topshiriq raqami
3. Topshiriqni bajarishda foydalanilgan funksiyalar tavsifi.
4. Topshiriq kodi.
5. Topshiriq natijasi (bu laboratoriya ishi uchun grafik, shakl va animasiya).
6. Dasturni elektron versiyasi (Albatta bu dasturlar kompyuterda o'qituvchiga ko'rsatiladi. O'qituvchini talab va takliflariga ko'ra dastur to'g'rilanadi va so'ngra hisobot tayyorlanadi.).

NAZARIY QISM

O'tish operatori goto. O'tish operatorining ko'rinishi:

goto <identifikator>. Bu operator identifikator bilan belgilangan operatorga o'tish kerakligini ko'rsatadi.

Misol uchun goto A1;...;A1:y = 5;

Tuzilmali dasturlashda goto operatoridan foydalanmaslik maslahat beriladi.

Lekin ba'zi hollarda o'tish operatoridan foydalanish dasturlashni osonlashtiradi.

Shartli operator. Shartli operator ikki ko'rinishda ishlatilishi mumkin:

if (ifoda) 1- operator ;**else** 2- operator;

yoki

if (ifoda) 1-operator;

Shartli operator bajarilganda avval ifoda hisoblanadi; agar qiymat rost ya'ni noldan farqli bo'lsa 1- operator bajariladi. Agar qiymat yolg'on ya'ni nol bo'lsa va else ishlatilsa 2-operator bajariladi. Operator else qismi har doim eng yaqin if ga mos qo'yiladi.

if(n>0)

if(a>b)

Z = a;

else

Z = b;

Agar else qismni yuqori if ga mos qo'yish lozim bo'lsa, figurali qavslar ishlatish lozim.

if(n>0) {

if(a>b)

z = a; }

else

z = b;

Misol tariqasida uchta berilgan sonning eng kattasini aniqlash dasturi:

```
#include<stdio.h>
```

```
int main()
```

```

{
float a,b,c,max;
scanf("%f",&a);
scanf("%f",&b);
scanf("%f",&c);
if (a>b)
if (a>c) max = a; else max = c;
else
if (b>c) max = b; else max = c;
printf("\n max = %f", max);
return 0;
}

```

Masalan: quyidagi ifodaning qiymatini hisoblash dasturi tuzilsin:

$$Y = \begin{cases} \sin x, & \text{agar } x < 5 \\ \sqrt[3]{x^2}, & \text{agar } x \geq 5 \end{cases}$$

```

#include <iostream.h>
#include <math.h>
void main ( )
{ float x, y;          // x va u ning toifasi haqiqiy
  cin >> x;           // x ning son qiymati kiritiladi
  if (x<5)            // agar x<5 bo'lsa
  { y=sin(x); goto cc; }      // 1-funksiya ishlaydi
  y=pow(x, 2/3.);      // aks holda 2-funksiya ishlaydi
cc: cout << "y=" << y << endl; // u ning javobi beriladi. cc-belgi
  }                    // main funksiyasi berkitildi.

```

yoki

```
# include <iostream.h>
# include <math.h>
void main ( )
{ float x, y;
  cin >> x;
  if (x<5) y=sin(x); else y=pow(x, 2/3.);
  cout << "y="<< y<< endl; }
```

Kalit bo'yicha tanlash operatori. Kalit bo'yicha tanlash switch operatori umumiy ko'rinishi quyidagicha:

```
switch(<ifoda>) {
  case <1-qiymat>:<1-operator> ; break;
  ...
  ...
  case < n -qiymat>: <n-operator>; break;
  default: <operator> }
```

Oldin qavs ichidagi butun ifoda hisoblanadi va uning qiymati hamma variantlar bilan solishtiriladi. Biror variantga qiymat mos kelsa shu variantda ko'rsatilgan operator bajariladi. Agar biror variant mos kelmasa default orqali ko'rsatilgan operator bajariladi. Uzish break operatori ishlatilmasa shartga mos kelgan variantdan tashqari keyingi variantdagi operatorlar ham avtomatik bajariladi. Quyidagi default, break va belgilangan variantlar ixtiyoriy tartibda kelishi mumkin. Umuman default yoki break operatorlarini ishlatish shart emas. Belgilangan operatorlar bo'sh bo'lishi ham mumkin.

Misol tariqasida bahoni son miqdoriga qarab aniqlash dasturini ko'ramiz.

```
#include <stdio.h>
int main()
{
  int baho;
  scanf("%d", &baho);
```

```

switch(baho)
{
case 2:printf("\n yomon");break;
case 3:printf("\n o'rta");break;
case 4:printf("\n yahshi");break;
case 5:printf("\n alo");break;
default: printf("\n noto'g'ri kiritilgan");
};
return 0;
}

```

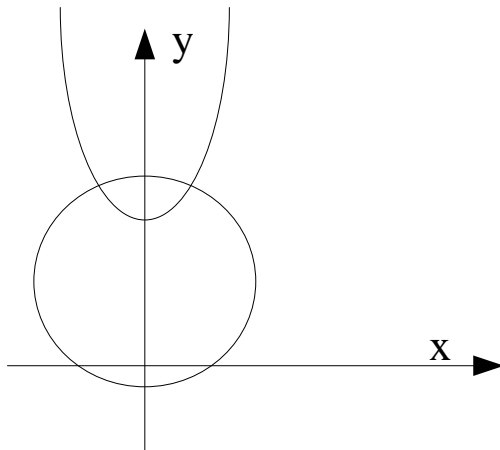
Keyingi misolda kiritilgan simvol unli harf ekanligi aniqlanadi:

```

#include <stdio.h>
int main()
{
char c;
scanf("%c", &c);
switch(c)
{
case 'a':
case 'u':
case 'o':
case 'i':
printf("\n Simvol unli");break;
default: printf("\n Simvol unli emas");
};
return 0;
}

```

Mantiqiy masala: ixtiyoriy berilgan $M(x,u)$ nuqta $u=x^2$ va $x^2+u^2=4$ aylana bilan kesishgan sohaga yoki shu aylananing 4-choragi tashqarisiga tushishini tekshiring.



Demak:

$$y > x^2 \text{ and } x^2 + y^2 \leq 4$$

$$\text{or } x > 0 \text{ and } y < 0 \text{ and}$$

$$x^2 + y^2 \geq 4$$

$$x = 1, y = 1 \rightarrow \text{false}$$

$$x=1, y=0 \rightarrow \text{true}$$

$$x = -2, y = 0.5 \rightarrow \text{false}$$

```
# include <iostream. h>
```

```
void main ( )
```

```
{ float x, y; int n;
```

```
cout << "nuqtaning koordinatalarini kiriting:";
```

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

```
if ((y>=x*x && x*x+y*y<=4) ||
```

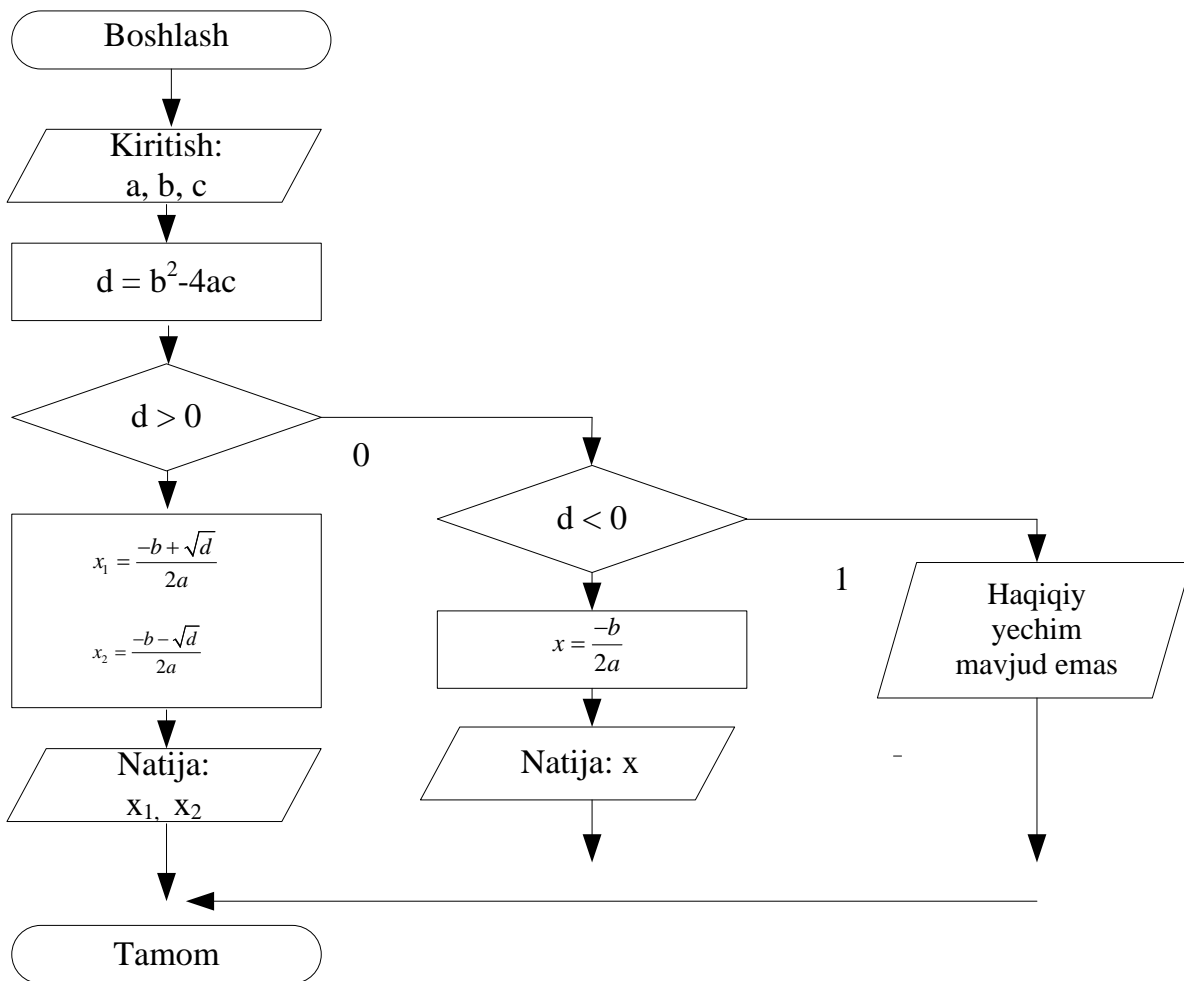
```
(x>0 && y<0 && x*x+y*y>=4)) n=1;
```

```
else n=0;
```

```
cout << "x="<<x<<endl;
```

```
cout << "y="<<y<< endl; cout << "n="<<n<< endl; }
```

3-misol: $ax^2+bx+c=0$ kurinishdagi kvadrat tenglamaning haqiqiy yechimlarini topish algoritmini tuzing. ($a \neq 0$; $b \neq 0$; $c \neq 0$;))



```

#include <iostream.h>
#include <math.h>
void main ( )
float a, b, c, d, x, x1, x2;
cout << "Tenglamaning
koeffitsiyentlarini kiriting: ";
cin >> a>>b>>c;
d = b*b - 4*a*c;
if ( d == 0 ) { x=- b / (2*a);
cout <<"x="<<x<<endl; goto
b15; }
if ( d > 0 )
{ x1 = (- b + sqrt(d)) / (2*a);
x2 = (- b - sqrt(d)) / (2*a);

```

```

#include <iostream.h>
#include <math.h>
void main ( )
float a, b, c, d, x, x1, x2; int v;
cout << "Tenglamaning
koeffitsiyentlari:";
cin >> a>>b>>c;
d = b*b - 4*a*c;
if (d<0) v=0;
if (d == 0) v=1; else v=2;
switch ( v )
{ case 0: cout <<"yechimi
yo'q"<< endl; break;
case 1 : { x=- b / (2*a);

```


<pre> cout <<"x1="<<x1<<"x2="<<x2<<endl; } else cout <<"yechimi yo'q"<< endl; b15 : } </pre>	<pre> cout <<" x="<<x<<endl; } break; case 2: { x1 = (- b + sqrt(d)) / (2*a); x2 = (- b - sqrt(d)) / (2*a); cout <<"x1="<<x1<<"x2="<<x2<<endl; } break; } </pre>
--	--

Nazorat savollari

1. C++ tilida tarmoqlanish qanday rasmiylashtiriladi ?
2. Shartli operatorning qisqa ko'rinishi?
3. Shartli operatorning uzun ko'rinishi?
4. Tanlash operatori?

BAJARISH UCHUN TOPSHIRIQLAR

№	Quyidagilarni hisoblash dasturi tuzilsin:
1	$y = \begin{cases} e^{\ln^2 x}, & \text{agar } x > 1 \\ \arctg \frac{2x}{z}, & \text{agar } x = 1 \\ \sqrt{ x - 0,5z }, & \text{agar } x < 1 \end{cases} .$
2	$z = \begin{cases} \sqrt{2x^3 + 3\ln 5x}, & \text{agar } x > 0 \\ 2^{x-5} - \sin^2 x, & \text{agar } x = 0 \\ \arcsin x^2, & \text{agar } x < 0 \end{cases}$
3	$y = \begin{cases} \operatorname{ctg}^2 x + \sqrt[3]{x+1}, & \text{agar } x > 0 \\ x^3 - e^{\operatorname{ctg} x} , & \text{agar } x = 0 \\ \ln x+0,5 , & \text{agar } x < 0 \end{cases}$
4	$z = \begin{cases} \ln 2,6x + 4,5 \operatorname{ctg}^2 x, & \text{agar } x > 0,5 \\ \sin^2 x + \sqrt{x+0,6}, & \text{agar } \tilde{\alpha} = 0,5 \\ \sin(x + 0,5x^2 + a \cdot b \cdot c), & \text{agar } x < 0,5 \end{cases}$ <p>bu yerda $\tilde{\alpha} = 5; b = -0,6; c = -7.$</p>

5	$y = \begin{cases} 4,3^{0,2x} \cdot \sin \sqrt{x} & , \text{agar } x > 1 \\ 4,17 \cdot \operatorname{arctg} x-5 & , \text{agar } x = 1 \\ \operatorname{ctg}(x+0,16) & , \text{agar } x < 1 \end{cases}$
6	$y = \begin{cases} \ln^2 x^2 + \frac{0,6x}{\sqrt{x+0,5}} & , \text{agar } x > 1 \\ (x+2)^2 + \operatorname{ctg} x-3 & , \text{agar } x = 1 \\ \sqrt[3]{ x+\sin x } & , \text{agar } x < 1 \end{cases}$
7	$z = \begin{cases} \ln \operatorname{arctg} x + 0,7 & , \text{agar } x > 0 \\ \operatorname{arcsin} x & , \text{agar } x = 0 \\ e^{\operatorname{arctg} x} & , \text{agar } x < 0 \end{cases}$
8	$z = \begin{cases} x^\alpha + e^{x-5} + \operatorname{ctg} 3x & , \text{agar } x > 0,6 \\ 4,14 \cdot \operatorname{ctg}^2 x^3 & , \text{agar } x = 0,6 \\ \frac{x+1}{\sqrt{ x }} & , \text{agar } x < 0,6 \end{cases}$ <p style="text-align: center;">bu yârda $\alpha = 5$.</p>
9	$y = \begin{cases} 2,6^{\ln x} - \sin \sqrt{x} & , \text{agar } x > 0,8 \\ a^{2x-\sqrt{b}} - \operatorname{arccos} x & , \text{agar } x = 0,8 \\ \cos 2x + x-a \cdot b & , \text{agar } x < 0,8 \end{cases}$ <p style="text-align: center;">bu yerda $a = 5$; $b = 4$.</p>
10	$y = \begin{cases} 2^{x+7} - 0,5 \ln(x+1) & , \text{agar } x > 1 \\ e^{\operatorname{arctg} x} - 5x^2 & , \text{agar } x = 1 \\ \frac{x-5}{2} + \operatorname{tg} x-3 & , \text{agar } x < 1 \end{cases}$
11	$y = \begin{cases} \sin(\ln x) + \sqrt[3]{x+5} & , \text{agar } x > 1 \\ 0,6 \cos 2x + 4,8^a & , \text{agar } a = 1 \\ a^2 - e^x + x + \ln^2 x & , \text{agar } x < 1 \end{cases}$ <p style="text-align: center;">bu yerda $a = 5,6$.</p>
12	$y = \begin{cases} 4,3^2 \cdot \operatorname{ctg} \sqrt{x} + e^{5x} & , \text{agar } x > 1 \\ \sqrt[3]{x^2+5} & , \text{agar } x = 1 \\ 2x + \operatorname{arctg} x^2 & , \text{agar } x < 0 \end{cases}$

13	$y = \begin{cases} x^2 + 4x - \pi \cdot x & , \text{ agar } x < 0 \\ (x^2 + 4)^2 - \sqrt{x^2 + 0,36} & , \text{ agar } 0 \leq x \leq 1 \\ x \cdot (x^2 + 3) + \ln^2(\pi + x) & , \text{ agar } x > 1 \end{cases}$
14	$y = \begin{cases} e^{x+0,6} - x-5 & , \text{ agar } x \geq 5 \\ \ln^2\left(1 + \frac{1}{x}\right) & , \text{ agar } 0 < x < 5 \\ ctg x + tg x & , \text{ agar } x \leq 0 \end{cases}$
15	$y = \begin{cases} e^{\ln^2 x} - tg^2 \frac{x}{0,5} & , \text{ agar } x = 1,2 \\ \sqrt{3x^2 + 9,36x + 5} & , \text{ agar } x > 1,2 \\ \ln 4x - 8,16 & , \text{ agar } x < 1,2 \end{cases}$
16	$w = \begin{cases} e^{\arctg x^2} + \sqrt{x^2 + 1} & , \text{ agar } x \geq 0,68 \\ \sqrt{x^2 + \ln x} & , \text{ agar } 0 < x < 0,68 \\ ctg 6x + 5 \ln x & , \text{ agar } x \leq 0 \end{cases}$
17	$y = \begin{cases} \cos^3 x & , \text{ agar } x < \frac{\pi}{2} \\ 1 - e^{\cos x} & , \text{ agar } x > \frac{\pi}{2} \\ \sin^2 x & , \text{ agar } x = \frac{\pi}{2} \end{cases}$
18	$y = \begin{cases} a^{2x-1} - \arccos x & , \text{ agar } x < 1 \\ \sin \cdot \ln(x^2 + 1) & , \text{ agar } x = 1 \\ \sqrt[4]{ x+5 } & , \text{ agar } x > 1 \end{cases}$ <p style="text-align: center;">bu yerda $a = 5$.</p>
19	$y = \begin{cases} \sqrt{2x+5,46} + e^{\sin x} & , \text{ agar } x > 1 \\ \ln \sin(x^3 + 1) & , \text{ agar } x = 1 \\ e^{x+t} + x+t & , \text{ agar } x < 1 \end{cases}$ <p style="text-align: center;">bu yerda $t = -0,5$.</p>

20	$y = \begin{cases} \sin^2 x + x-1 + 2 \cdot \sqrt{x \cdot s} & , \text{ agar } x > 1,5 \\ \sqrt{3a - 2bx + x^2} & , \text{ agar } x = 1,5 \\ e^{\operatorname{arctg} x} & , \text{ agar } x < 1,5 \end{cases}$ <p style="text-align: center;">bu yerda $a = 10$; $b = 1,5$.</p>
21	$y = \begin{cases} \operatorname{ctg} \frac{3x}{5} + e^{\sqrt{\operatorname{tg} x}} & , \text{ agar } x > 2,5 \\ \ln 2x^3 + \sqrt{x+1} & , \text{ agar } x = 2,5 \\ x^3 - \operatorname{ctg} x & , \text{ agar } x < 2,5 \end{cases}$
22	$z = \begin{cases} \operatorname{arctg} x^2 + e^{\operatorname{tg} x} & , \text{ agar } x > 0 \\ \sin x + \ln x+5 & , \text{ agar } x < 0 \\ \cos(x+4,5a) & , \text{ agar } \tilde{\sigma} = 0 \end{cases}$ <p style="text-align: center;">bu yerda $a = -0,5$.</p>
23	$q = \begin{cases} 2 \operatorname{tg} x^2 + e^{\sin x} & , \text{ agar } 0 \leq x \leq 1 \\ 4 \operatorname{ctg} 2x + \ln 2,6x & , \text{ agar } x > 1 \\ \operatorname{arcctg} x & , \text{ agar } x < 0 \end{cases}$
24	$p = \begin{cases} \sqrt{x^2 + 3 \ln 5x} & , \text{ agar } x > 6,139 \\ x^2 + 5x - 5 & , \text{ agar } x < -1,5 \\ \frac{ x - 8,16 }{x+4} - 5x & , \text{ agar } -1,5 \leq x \leq 6,139 \end{cases}$
25	$y = \begin{cases} \sqrt{x} + \cos x + 0,5 \cdot x^2 & , \text{ agar } x > 1,5 \\ b \cdot e^{x+9} - \cos x & , \text{ agar } 0 \leq x \leq 1,5 \\ \operatorname{tg} x^2 + x^{a+b} - x^2 & , \text{ agar } x < 1,5 \end{cases}$ <p style="text-align: center;">bu yerda $\alpha = 3$; $b = -5$.</p>
26	$y = \begin{cases} \frac{a}{a^2 + x} & , \text{ agar } a > 5 \\ \frac{1}{2a^3 + \sin a} & , \text{ agar } a \leq 5 \end{cases}$
27	$y = \begin{cases} \sqrt{k} & , \text{ agar } \sin k \leq 0,2 \\ \frac{1}{\sqrt{k}} & , \text{ agar } \sin k > 0,2 \end{cases}$

28	$y = \begin{cases} \sin^2(2x) - \cos^2 x & , \text{ agar } x > 0,3 \\ \frac{1}{x^2 - \sqrt{x}} & , \text{ agar } x \leq 0,3 \end{cases}$
29	$p = \begin{cases} \frac{1}{2}(3x^2 - 1) & , \text{ agar } x > 0,4 \\ \frac{1}{2}(5x^3 - 3x) & , \text{ agar } x \leq 0,4 \end{cases}$
30	$y = \begin{cases} x^2 + 4 & , \text{ agar } x < 10 \\ x^3 - 7 & , \text{ agar } x \geq 10 \end{cases} \quad \text{bu yerda } x = \frac{a^2 - b}{c}$

6-Laboratoriya ishi. Takrorlanuvchi jarayonlarni dasturlash

Ishning maqsadi

1. Oldshartli takrorlanish bilan tanishish.
2. So'ngshartli takrorlanish bilan tanishish.
3. Parametrli takrorlanish bilan tanishish.
4. Ichma-ich joylashgan takrorlanish bilan ishlash.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarning dasturini tuzishi lozim.
2. Hisobot shaklida oldin vazifa, uni bajarishda foydalanilgan funksiyalar, dastur kodi va natijalarni keltirib o'tishi lozim.
3. 6-laboratoriya ishidagi berilgan barcha vazifalarni bitta hisobot shaklida topshiring.

NAZARIY QISM

Oldshartli while operatori. Oldingi shartli while operatori quyidagi umumiy ko'rinishga egadir:

while(ifoda)

Operator;

Bu operator bajarilganda avval ifoda hisoblanadi. Agar uning qiymati 0 dan farqli bo'lsa operator bajariladi va ifoda qayta hisoblanadi. To ifoda qiymati 0 bo'lmaguncha sikl qaytariladi.

Agar dasturda while (1); satr qo'yilsa bu dastur hech qachon tugamaydi.

Misol. Berilgan n gacha sonlar yig'indisini hisoblash dasturi.

```
#include <stdio.h>
```

```
void main()
```

```
{
```

```
long n,i = 1,s = 0;
```

```
scanf("%d",&n);
while (i<= n )
    s+ = i++;
printf("\n s = %d",s);
}
```

Bu dasturda $s+ = i++$ ifoda $s = s+i$; $i = i+1$ ifodalarga ekvivalentdir.

Quyidagi dastur to nuqta bosilmaguncha kiritilgan simvollar va qatorlar soni hisoblanadi:

```
#include <stdio.h>
int main()
{
    int nc = 0, nl = 0;
    char c;
    while ((c = getchar()) != '.')
    {
        ++nc;
        if (c == '\n') ++nl;
    };
    printf("satrlar = %d simvollar = %d \n",nl,nc);
    return 0;
}
```

So'ngshartli do-while operatori. So'ngshartli do-while operatori umumiy ko'rinishi quyidagicha:

```
do
    Operator
while(ifoda);
```

Sikl operatorining bu ko'rinishida avval operator bajariladi so'ngra ifoda hisoblanadi. Agar uning qiymati 0 dan farqli bo'lsa operator yana bajariladi va hokazo. To ifoda qiymati 0 bo'lmaguncha sikl qaytariladi.

Misol. Berilgan n gacha sonlar yig'indisi.

```
#include <stdio.h>  
int main()  
{  
long n,i = 1,s = 0;  
scanf("%d",&n);  
do  
s+ = i++;  
while (i< = n);  
printf("\n s = %d",s);  
return 0;  
}
```

Bu dasturning kamchiligi shundan iboratki agar n qiymati 0 ga teng yoki manfiy bo'lsa ham, sikl tanasi bir marta bajariladi va s qiymati birga teng bo'ladi.

Parametrlil for operatori. Parametrlil for operatori umumiy ko'rinishi quyidagicha:

```
for( 1-ifoda;2- ifoda; 3-ifoda)
```

```
Operator(lar);
```

Bu operator quyidagi operatorga mosdir.

```
1-ifoda;
```

```
while(2-ifoda) {
```

```
operator
```

```
3-ifoda
```

```
}
```

1-misol. Berilgan n gacha sonlar yig'indisi.

```
#include <stdio.h>
```

```
int main()
```



```

{
int n;
scanf("%d",&n);
int s = 0;
for(int i = 1;i<= n; i++) s+ = i;
printf("\n%d",s);
return 0;
}

```

Siklda bir nechta schyotchikni qo'llanilishi. Parametrli for siklining sintaksisi unda bir nechta o'zgaruvchi - schyotchikni qo'llanilishiga, siklni davom etishini murakkab shartlarini tekshirishga va sikl schyotchiklari ustida ketma-ket bir nechta operatsiyani bajarilishiga imkon beradi.

Agarda bir nechta schyotchikka qiymat o'zlashtirilsa yoki ular o'rtasida bir nechta operatsiya bajarilsa, bu ifodalar vergul bilan ajratilgan holda ketma – ket yoziladi.

for siklida bir nechta schyotchikni qo'llanilishi:

```

#include <stdio.h>
#include<conio.h>
int main()
{
int i,j;
for (i = 0, j = 0; i<3; i++, j++)
printf("i:%d j:%d\n",i,j);
getch();
return 0;
}

```

Hatija:

```

i: 0      j: 0
i: 1      j: 1
i: 2      j: 2

```

Uzish break operatori. Ba'zi hollarda sikl bajarilishini ixtiyoriy joyda to'xtatishga to'g'ri keladi. Bu vazifani break operatori bajarishga imkon beradi. Bu operator darhol sikl bajarilishini to'xtatadi va boshqaruvni sikldan keyingi operatorlarga uzatadi.

Misol:

```
#include <stdio.h>  
int main()  
{  
int n;  
while(1)  
{  
scanf("%d",&n);  
if(n == 1||n == 0) break;  
}  
printf("Sikl tugadi");  
return 0;  
}
```

Bu misolda while(1) operatori yordamida cheksiz sikl hosil qilinadi. Agar 1 yoki 0 soni kiritilsa sikl to'xtatiladi.

Qaytarish continue operatori. Sikl bajarilishiga ta'sir o'tkazishga imkon beradigan yana bir operator continue operatoridir. Bu operator sikl qadamini bajarilishini to'xtatib for va while da ko'rsatilgan shartli tekshirishga o'tkazadi.

Misol:

```
#include <stdio.h>  
int main()  
{  
int n;  
for(;;)  
{  
scanf("%d",&n);
```

```

if(n == 1||n == 0) continue;
break;
}
printf("Sikl tugadi");
return 0;
}

```

Bu misolda for(;;) operatori yordamida cheksiz sikl hosil qilinadi. Agar 1 yoki 0 sonlardan farqli son kiritilsa sikl to'xtatiladi.

2-misol: $b = 2*(a+5)$; $a \in [1, 10]$; $h=1$;

```
# include <iostream.h>
```

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

```
void main ( )
```

```
{
```

```
int a=1, b;
```

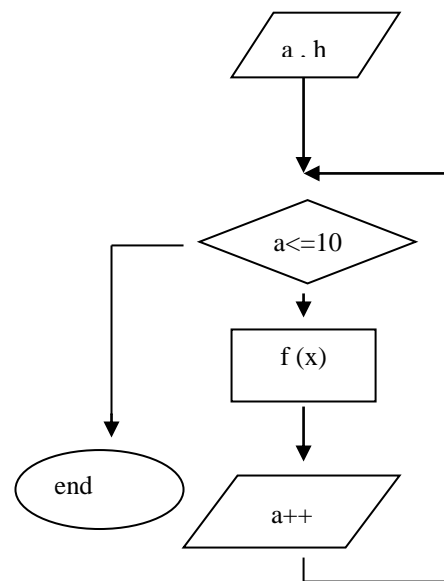
```
while (a<=10)
```

```
{ b = 2*(a+5); cout << "b=" <<b;
```

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

```
a++ ; }
```

```
}
```



3-misol: $y=\sin x$; $x \in [1,2]$; $h=0.1$

```
# include <iostream.h>
```

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

```
void main ( )
```

```
{ float x=1, y;
```

```
do
```

```
{ y=sin(x); cout << "x=" <<x<< " y=" <<y<<endl; x+ = 0.1; }
```

```
while (x<=2); getch(); }
```

4-misol. $Y = \cos x$; $x \in [2,3]$; $h=0,2$;

```
# include <iostream.h>
```

```
# include <math.h>

void main ( )
{ float x, y;
  for (x=2; x<=3; x+ = 0.2)
  { y=cos(x); cout << "x="<< x << " y="<< y << endl; }
}
```

5-misol. 100 gacha bo'lgan juft sonlarni ekranga chiqarish dasturi.

```
# include <iostream.h>

void main ( )
{ int i = 2;
  while (i<=100)
  { cout << "i="<< i; i += 2; } }
```

<pre>.... for (int i=2; i<=100; i +=2) cout << "i="<< i;</pre>	<pre>do cout << "i="<< i; i += 2; while (i<=100);</pre>
---	--

6-misol : 1 dan 100 gacha bo'lgan 3 raqami bilan tugaydigan sonlarni ekranga chiqarish dasturini tu'zing (2 xil usulda).

<pre>..... int i=3; while (i <=100) { cout << "i="<< i; i += 10; }</pre>	<pre>..... for (i = 3;i <= 100; i + = 10) cout << "i="<< i;</pre>
---	--

7-misol. Qadimiy masala. Bir odam 100 so'm bilan bozorga bordi. Bozorda 1ta sigir 10 so'm, 1ta qo'y 3 so'm, 1ta echki 0.5 so'm va harid qilingan qoramollarning umumiy soni 100ta bo'lsa, nechta sigir, qo'y va echki sotib olindi?

Sigirlar soni: x, qo'ylar soni u, echkilar soni z deb olinsin.

<pre># include <iostream.h> int main () { int x, y, z, s; for (x=1; x<=100; x++) for (y=1; y<=100; y++) if (19*x + 5*y == 100) { z = 100 - x - y; cout << "x="<<x; cout << "y="<<y; cout << "z="<<z; } return 0; }</pre>	<pre># include <iostream.h> int main () { int x, y, z, s; for (x=1; x<=100; x++) for (y=1; y<=100; y++) for (z=1; z<=100; z++) if (x + y + z == 100) { cout << "x="<<x; cout << "y="<<y; cout << "z="<<z; } return 0; }</pre>
---	---

Nazorat uchun savollar

1. Takrorlanish turlarini ko'rsating.
2. Oldshartli takrorlanish haqida tushuncha bering?
3. So'ngshartli takrorlanish haqida tushuncha bering?
4. Parametrli takrorlanish haqida tushuncha bering?

BAJARISH UCHUN TOPSHIRIQLAR

6.1-jadval

№	Variantlar
1	$S = \frac{a+b}{2} \sum_{i=1}^4 \prod_{j=1}^5 \frac{\ell n(i^2+1)}{2 \cdot i \cdot j}, \text{ bu yerda } a=5; b=6.$
2	$S = \frac{b-a}{2a} \sum_{k=1}^5 \sum_{\ell=1}^3 \frac{e^{k-\ell}}{\sqrt{k+\ell}}, \text{ bu yerda } a=-3,5; b=3.$

3	$P = \prod_{n=1}^3 \sum_{m=1}^5 \frac{\sin(n+m^2)}{\sqrt{n+\frac{m}{2}}}$
4	$S = a \cdot \sum_{k=1}^3 \prod_{\ell=1}^2 \frac{2,5^{k+\ell}}{\sqrt[3]{k+\ell}}, \text{ bu yerda } a = 5,9.$
5	$P = \frac{6,3}{2^x} \prod_{i=1}^4 \prod_{j=1}^5 \frac{2,5^{i+j}}{i+j}, \text{ bu yerda } \tilde{\sigma} = 3.$
6	$S = \sum_{k=1}^{10} \prod_{\ell=1}^5 \frac{\ln(k^2+\ell)}{4,5^{k+\ell}}$
7	$S = a^{\sqrt{x}} \cdot \sum_{i=1}^5 \sum_{j=1}^6 \frac{\operatorname{ctg} ij}{5ij}, \text{ bu yerda } a = 3; x = 3.$
8	$P = \prod_{i=1}^4 \ln i + \prod_{j=1}^5 \sin j$
9	$S = \ln a \cdot \sum_{i=1}^5 \frac{e^{i-1}}{5i} + \sum_{j=1}^3 \ln j, \text{ bu yerda } a = 3,65.$
10	$P = e^{\sqrt[3]{x-3}} \prod_{t=1}^6 \prod_{q=1}^7 \frac{t+5}{\ln q}, \text{ bu yerda } x = 6,59.$
11	$P = \prod_{k=1}^5 \prod_{l=1}^8 \frac{\arccos(k+l)}{\sqrt{k^4+l^4}}$
12	$S = \sum_{k=1}^9 \sum_{\ell=1}^3 \frac{2,5^{k+\ell}}{\sqrt[3]{k^\ell}},$
13	$S = \sum_{n=1}^5 \sum_{m=1}^6 \frac{\sin(n+m^{\sqrt{x}})}{\sqrt{m+\frac{x}{n}}}, \text{ bu yerda } x = 18,61.$
14	$Q = \ln n^2 a \sum_{k=1}^7 \prod_{\ell=1}^5 \frac{e^k - e^{-\ell}}{\sqrt{k+\ell^2}}, \text{ bu yerda } a = 10,89.$

15	$W = (b^2 - 4, 15) \sum_{k=1}^6 \prod_{\ell=1}^3 \frac{t g^2(k + \ell)}{k^3 + \ell}, \text{ bu yerda } b = 3,09.$
16	$Z = \sum_{i=1}^9 \prod_{j=1}^{10} \frac{\sin(i^2 + j)}{2^{i+j}}$
17	$P = 2a \sum_{i=1}^5 ct gi + 5b \prod_{j=1}^5 tg j, \text{ bu yerda } a = 1,5; b = 5.$
18	$A = \sum_{k=1}^4 \prod_{\ell=1}^5 \frac{\ell n^3(k + \ell)}{5a}, \text{ bu yerda } a = 3,5.$
19	$Q = e^{2x+1} \prod_{i=1}^9 \frac{i+1}{5} + e^{\ell n x} \sum_{j=1}^5 \frac{j-1}{5}, \text{ bu yerda } x = 4,5.$
20	$S = \sum_{k=1}^7 \prod_{t=1}^3 \frac{\ell n^3 a^t}{2,61^{k+t}}, \text{ bu yerda } a = 10,719.$
21	$Y = \frac{a+b}{5} \prod_{j=1}^6 \frac{j+1}{2} + \sum_{i=1}^3 \ell ni, \text{ bu yerda } a = -5; b = 3,5.$
22	$P = \prod_{k=1}^3 \prod_{j=1}^4 \frac{\arctg(k + \pi \cdot j)}{k^2 + 5}$
23	$S = \sum_{n=1}^6 \prod_{m=1}^4 \frac{\ell n(n^2 + m)}{4^{\sqrt{n+m+0,6}}}$
24	$Z = tg \frac{2a}{3} \sum_{n=1}^3 \prod_{m=1}^5 \frac{\sqrt[3]{n+m^2}}{\sqrt{ n-m }}, \text{ bu yerda } a = 0,3.$
25	$W = z^2 \sum_{n=1}^5 \prod_{\ell=1}^4 \frac{t g^2(k + \ell)}{2,5^{n+\ell}}, \text{ bu yerda } z = 5.$
26	$S = \sum_{x=1}^{10} \left(\frac{\sqrt{ x-2,1 } - \sqrt[3]{x}}{z + \frac{x^2}{2} + \frac{x^3}{3}} + \sum_{y=4}^9 \frac{1}{x} \left(\arctan\left(\frac{z}{y}\right) + \sin\left(\frac{1}{y} + \frac{\pi}{6}\right) \right) \right) \text{ bu yerda } z = 1.05$

27	$P = \prod_{l=2}^5 \left(\frac{\log_4 \left(x + \frac{1}{l} \right) + \sqrt[3]{\frac{x}{l}}}{ x^2 + y^2 + \sqrt[3]{\frac{1}{l}}} + \sum_{k=3}^7 \frac{\arctan \left(\frac{l+k}{x} \right)}{\log_5 \left(\frac{2*k+l^2}{2} \right) + y^2 + 1} \right)$ <p>bu yerda $x = 1.5; y = 0.94$</p>
28	$U = \sum_{x=1}^5 \left(\frac{1 + \cos^2 \left(\arcsin \left(\frac{y}{x} \right) \right)}{e^{1 + \frac{1}{x}} - \sin^2(y)} - \prod_{z=3}^6 \frac{\cos^2 \left(\frac{x}{5*z} \right) + \sin^2 \left(\frac{\pi}{2} - \frac{z*x}{10} \right)}{1 + \sqrt[5]{y^2 + \frac{1}{x^2 + z^2}}} \right)$ <p>bu yerda $y = 0.72$</p>
29	$S = \sum_{k=1}^4 \left(\frac{ \cos(y) + \cos(z) ^{1 + \frac{1}{k^2}}}{\sqrt{1 + \frac{1}{k^2} + \frac{1}{k^4} + \frac{1}{k^6}}} \cdot \sum_{i=3}^7 \frac{y^z + \sin^2 \left(\frac{k*i}{i} \right)}{e^{0.2*i} + \sqrt[4]{1/k*i}} \right)$ <p>bu yerda $z = 1.1; y = 1.1$</p>
30	$F = \prod_{y=1}^4 \left(\frac{\frac{\pi}{2} - \arctan \left(\frac{x}{y} \right)}{\sqrt[4]{y^2 + x^2} + 1} + \prod_{z=2}^6 \frac{\arcsin \left(\frac{y}{2*z} \right) + e^{-zx}}{ x-z \sqrt[4]{y}} \right) \text{ bu yerda } x = 0.86$

6.2-jadval

N _o	Variantlar
1	a) $\sum_{n=1}^{10} \frac{1}{n^3}$; b) $\sum_{R=1}^{15} \frac{R^3}{R^4 + 3R^2 + e^{-R}}$; c) $\prod_{R=1}^{15} \prod_{i=1}^{10} \frac{R^i + 1}{R^4 + 3^i * R + e^{-R}}$
2	a) $\sum_{n=1}^{10} \frac{2}{n^3(n+1)}$; b) $\sum_{R=1}^{14} \frac{R^2 + R-2 }{\ln R + 3R}$; c) $\sum_{R=1}^{14} \sum_{m=1}^4 \frac{R \cdot m + R^{-m} + 2 }{\ln R + 3m}$

3	a) $\sum_{n=1}^{20} \frac{3}{(2n+1)^3}$; b) $\sum_{R=1}^{17} \frac{R+1}{\sin R + e^{-R} + 1}$; c) $\prod_{R=1}^{16} \sum_{i=1}^6 \frac{R+3}{R^3 + 3R + i^3}$
4	a) $\sum_{R=1}^{13} \frac{4}{R(R+1)}$; b) $\sum_{R=1}^{10} \frac{R^{R+1}}{2^{R+1} + (R+1)^4}$; c) $\sum_{R=1}^{10} \prod_{i=1}^{10} \frac{(R+1)^i + 4}{(-1)^R + 3(-1)^i + i^R}$
5	a) $\sum_{m=1}^{10} \frac{5}{m^2 + m + 4}$; b) $\sum_{R=1}^{15} \frac{(100-R)^2}{\lg R + 5^{-R}}$; c) $\sum_{i=1}^{13} \sum_{R=1}^4 \frac{(-1)^i \cos(i+R) + 5}{5i + 7^{-R} + i^{-R}}$
6	a) $\prod_{n=1}^8 \frac{n+6}{n^2 + 4n + 1}$; b) $\sum_{i=1}^{17} \frac{i+6}{i^4 + 27i + 7}$; c) $\prod_{R=1}^8 \prod_{i=1}^{14} (-1)^i \frac{\sqrt{5i^4 + e^{-R} + 6}}{\cos(i+1)^3 - R^{-i}}$
7	a) $m! + 7$; b) $\sum_{i=1}^{10} \frac{(-1)^i \cdot 7^{-i}}{1+i+i^2}$; c) $\sum_{i=1}^{13} \sum_{R=1}^{14} \frac{(-1)^i \cos(i+R) + 5}{5i + 7^{-R} + i^{-R}}$
8	a) $(m+1)!$; b) $\sum_{n=1}^{12} \frac{10n-8}{10n^2 - 3n + 8}$; c) $\sum_{i=1}^{13} \sum_{m=2}^5 \left[\frac{i^m + 4m + e^m}{m^i} \right]$
9	a) $\prod_{n=1}^{10} \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}$; b) $\sum_{i=1}^7 (2i + 5i + 9)$; c) $\sum_{R=1}^{17} \prod_{m=1}^5 \sqrt{\frac{R + m^3 + e^{-m} + 9}{\text{Log}_m R + (mR)^3}}$
10	a) $\prod_{n=1}^{10} \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}$; b) $\prod_{n=1}^{10} \frac{n^2 + 3n + 10}{\sqrt[3]{n^2 + 7n + 91}}$; c) $\sum_{R=1}^{17} \sum_{m=1}^5 \sqrt{\frac{Tg(R+m)^2 + 10R}{R + m^{-R} + e^{m-R}}}$
11	a) $\prod_{n=1}^{10} \frac{n^2 + 9^{-n}}{e^{-n} + n^{n^{-n}}}$; b) $\prod_{n=1}^{10} \frac{1}{n^4 + 1}$; c) $\prod_{i=1}^{15} \sum_{m=1}^{11} \left[e^{\sqrt{i^2 + m^{1-i}}} + \frac{i^2 + 11}{m^4 + i^{-m}} \right]$
12	a) $\prod_{i=1}^9 \frac{i^4 + i^2 + 12}{\sqrt{i^3 + e^{-i}}}$; b) $\sum_{R=1}^{10} \frac{R+I}{R^5 + 5R + 1,2}$; c) $\sum_{m=1}^9 \prod_{n=1}^7 \sqrt{\frac{m^3 - n^2 + 3,4}{m^{-n} + m^{-m} + 12}}$
13	a) $\prod_{n=1}^{15} \frac{13}{n^3 + 5n + 7}$; b) $\sum_{m=2}^{13} \frac{(-1)^m \sqrt{m}}{2^{-2m}}$; c) $\prod_{i=1}^{21} \prod_{m=1}^{20} tg \frac{i^{-m} - i^{3-m} - i^2 + 1,3}{m^{-i} + m^{-6} + im + 13}$
14	a) $\sum_{R=1}^{19} \frac{R^2 + 14}{\sqrt{3^{-R} + R^3}}$; b) $\prod_{n=1}^{14} \frac{n+b}{n + \frac{1}{n}}$; c) $\sum_{i=1}^6 \sum_{m=1}^{14} \lg \frac{\sqrt[3]{m^2 + e^{m-i}}}{i^2 + 2^{i-m}}$
15	a) $\prod_{i=1}^{14} \frac{ i-15 + i^3}{\ln i + 7i}$; b) $\sum_{R=1}^{10} \frac{(-1)^R * (R+1)}{R^3 + R^2 + 1}$; c) $\prod_{n=1}^{14} \sum_{m=1}^{16} (-1) \frac{m \log_n (m+5) + 1,5}{2^{m-9} + (n+3)^{-m} + nm}$
16	a) $\sum_{i=-22}^{40} \frac{i \sqrt{ i-2i^3 + 16}}{\ln i+3 + 1,6}$; b) $\sum_{n=1}^{20} (-1)^n \frac{n+c}{2n^4 + 1}$; c) $\prod_{R=4}^{16} \prod_{m=1}^{17} \sqrt{\frac{R^m + 4R - m + 1,6}{\sin(m+R) - m}}$
17	a) $\prod_{R=1}^{17} \frac{R+17}{2R^2 + 9}$; b) $\sum_{R=1}^{13} (-1) \frac{R^R \sqrt{R+1} + R^2}{2R^2 + 4R + 11}$; c) $\sum_{m=1}^{17} \prod_{n=1}^{10} \sqrt{\frac{m^3 - n^2 + 1,7}{m^n + m^m + 12}}$

18	$\text{a) } \sum_{n=1}^{10} \frac{18}{5-17n+n^3}; \quad \text{b) } \prod_{m=-12}^0 \frac{m^2 \sqrt{ m +1,8}}{m^2+4m+(-1)^m} \quad \text{c) } \sum_{i=1}^{17} \prod_{R=1}^{10} \frac{\sqrt{e^{i+R}(i+R)^{i-R}}}{ 4i^3-R^4 }$
19	$\text{a) } \sum_{n=1}^9 \frac{19n}{3+n+n^2}; \quad \text{b) } \sum_{t=3}^9 \frac{\text{tg}(t+3)}{t^3+2t+e^{t-1}}; \quad \text{c) } \prod_{i=-4}^0 \prod_{m=2}^{19} \frac{(\sqrt{i}+m)}{(i+m)}$
20	$\text{a) } \prod_{n=1}^{20} (-1)^n \frac{1+n^2}{1+n^3}; \quad \text{b) } \sum_{m=10}^6 \frac{\text{sign}(m)}{\sqrt[4]{m^2+e^{i+13}}}; \quad \text{c) } \prod_{n=1}^{11} \prod_{R=2}^{16} \frac{n^3-R^2+20}{(n-R +n)^{-R}}$
21	$\text{a) } \prod_{n=1}^{15} (-1)^n \frac{n+21}{9+5n^3} \quad \text{b) } \sum_{R=1}^{12} \frac{2^{-R}+2^R+21}{R^2+e^{2-13}}; \quad \text{c) } \prod_{i=1}^{16} \sum_{R=1}^6 \frac{\text{sign}(\text{Sin}(i+R))}{(i+R)^{i-R}-21}$
22	$\text{a) } \sum_{n=1}^{10} \frac{3n^3+4n+18}{n^3+Ln(m+3)}; \quad \text{b) } \prod_{R=1}^{10} (-1)^R \frac{R+22}{R^3+7R+5} \quad \text{c) } \sum_{i=1}^{17} \prod_{R=1}^{10} \frac{Lni+R^2}{ 4i^3-R^4 }$
23	$\text{a) } \prod_{R=1}^7 \frac{61R+17}{2R^2+9,6}; \quad \text{b) } \sum_{R=1}^{12} (-1)^R \frac{\text{Arc cos}(R+1)+R^2}{2R^2+\text{Tg}R+11}; \quad \text{c) } \sum_{m=1}^{17} \prod_{n=1}^{10} \frac{(1-m)^{n-m}}{(m+n+5)^3}$
24	$\text{a) } \sum_{R=1}^{14} \frac{R+2,4}{R^2+7R+1}; \quad \text{b) } \prod_{q=1}^{18} (-1)^q \frac{\text{Cos}(q^2+5)}{q^4+ q-71 }; \quad \text{c) } \sum_{i=1}^{16} \prod_{m=1}^{13} \frac{\text{arctg}(i+m)}{Lni+0,24}$
25	$\text{a) } \sum_{R=1}^{15} \frac{3R-2,5}{\sqrt{R^2+3R+8}}; \quad \text{b) } \prod_{n=15}^{45} \frac{\text{arctgn}}{n^{1,6}-Ln(n+25)}; \quad \text{c) } \prod_{i=1}^{25} \sum_{m=3}^{12} \frac{(i^3+m^4)^{\frac{1}{i}}}{\sqrt{Ln(i+m)+i^m}}$
26	$\text{a) } \prod_{i=1}^{10} \frac{6i-2,6}{i^{4^i}-3i^3+i-1}; \quad \text{b) } \prod_{q=1}^{16} \frac{\text{Arc cos}(q^2+5)}{q^4+\text{Tg}(q+1)}; \quad \text{c) } \prod_{i=1}^{26} \sum_{R=1}^{26} \frac{\text{sign}(\ln(i+R))}{(i+R)^{i-R}}$
27	$\text{a) } \prod_{R=1}^{27} \frac{R+2,7}{2R^3+9}; \quad \text{b) } \sum_{R=1}^{13} (-1)^R \frac{\sqrt[R]{R+1+R^2}}{R^2+R+27}; \quad \text{c) } \prod_{R=4}^{16} \prod_{m=1}^{17} \frac{\sqrt{R^2+4R-m}}{\sin(m+R)-m^2}$
28	$\text{a) } \sum_{i=1}^{28} (4i-28)^2; \quad \text{b) } \prod_{n=1}^{11} \frac{n^3+3n+2,8}{\sqrt[3]{n^2+7n+91}}; \quad \text{c) } \sum_{R=1}^6 \sum_{m=1}^{13} \frac{\sqrt{(R+m)^2+28}}{R+m^2+2^{m-R}}$
29	$\text{a) } \sum_{m=1}^{10} \frac{29}{m^2+m+4}; \quad \text{b) } \prod_{R=1}^{25} \frac{(100-R)^2}{\lg R+2^{R-49}}; \quad \text{c) } \sum_{i=1}^8 \sum_{R=1}^{13} \frac{(-1)^i \cos(i+R)}{5i+7R+iR-29}$
30	$\text{a) } m!; \quad \text{b) } \sum_{n=1}^{30} \frac{10n-30}{10n^2-3n+8}; \quad \text{c) } \sum_{R=1}^6 \sum_{m=1}^{13} \frac{\sqrt{\text{tg}(R+m)^2+5R}}{R+m^3+2^{m-R}+30}$

7-Laboratoriya ishi. Massivlarni tashkil etish.

Ishning maqsadi

1. C++ da massivlarni tashkil etish bilan tanishish.
2. Bir o'lchamli massivlarga ishlov berish.
3. Ko'p olchamli massivlarni tashkil etish.
4. Massivlarga saralash usullarini qollash.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarning dasturini yozadi.
2. Har bir vazifani bajarishda foydalanilgan o'zgaruvchi va funksiyalar haqida hisobotda ko'rsatish lozim.
3. Hisobot shaklida oldin vazifa, uni bajarishda foydalanilgan funksiyalar, dastur matni va bajarishdan hosil bo'lgan natijani keltirish lozim.
4. 7-laboratoriya ishida berilgan barcha vazifalarni bitta hisobot shaklida topshiring.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. Laboratoriya ishidagi topshiriq raqami
3. Topshiriqni bajarishda foydalanilgan funksiyalar tavsifi.
4. Topshiriq kodi.
5. Topshiriq natijasi.
6. Dasturning elektron versiyasi (Albatta bu dasturlar kompyuterda o'qituvchiga ko'rsatiladi. O'qituvchini talab va takliflariga ko'ra dastur to'g'rilanadi va so'ngra hisobot tayyorlanadi.).

NAZARIY QISM

Massiv tushunchasi. Massiv bu bir turli nomerlangan ma'lumotlar jamlanmasidir. Massiv indeksli o'zgaruvchi tushunchasiga mos keladi. Massiv ta'riflanganda turi, nomi va indekslar chegarasi ko'rsatiladi. Masalan, type turidagi length ta elementdan iborat a nomli massiv shunday e'lon qilinadi:

```
type a[length];
```

Bu maxsus a[0], a[1], ..., a[length -1] nomlarga ega bo'lgan type turidagi o'zgaruvchilarning e'lon qilinishiga to'g'ri keladi.

Massivning har bir elementi o'z raqamiga - indeksga ega. Massivning x-nchi elementiga murojaat indekslash operatsiyasi yordamida amalga oshiriladi:

```
int x = ...; //butun sonli indeks
```

```
TYPE valuye = a[x]; //x-nchi elementni o'qish
```

```
a[x] = valuye; //x- elementga yozish
```

Indeks sifatida butun tur qiymatini qaytaradigan har qanday ifoda qo'llanishi mumkin: char, short, int, long. C da massiv elementlarining indeksleri 0 dan boshlanadi (1 dan emas), length elementdan iborat bo'lgan massivning oxirgi elementining indeksi esa - bu length -1 (length emas) ga teng. Massivning int z[3] shakldagi ta'rifi, int turiga tegishli z[0], z[1], z[2] elementlardan iborat massivni aniqlaydi.

Massiv chegarasidan tashqariga chiqish (ya'ni mavjud bo'lmagan elementni o'qish/yoziqshga urinish) dastur bajarilishida kutilmagan natijalarga olib kelishi mumkin. Shuni ta'kidlab o'tish lozimki, bu eng ko'p tarqalgan xatolardan biridir.

Agar massiv inisializasiya qilinganda elementlar chegarasi ko'rsatilgan bo'lsa, ro'yxatdagi elementlar soni bu chegaradan kam bo'lishi mumkin, lekin ortiq bo'lishi mumkin emas.

Misol uchun int a[5] = {2,-2}. Bu holda a[0] va a[1] qiymatlari aniqlangan bo'lib, mos holda 2 va -2 ga teng. Agar massiv uzunligiga qaraganda kamroq element berilgan bo'lsa, qolgan elementlar 0 hisoblanadi:

```
int a10[10] = {1, 2, 3, 4}; //va 6 ta nol
```

Agar nomlangan massivning tavsifida uning o'lchamlari ko'rsatilmagan bo'lsa, kompilyator tomonidan massiv chegarasi avtomatik aniqlanadi:

```
int a3[] = {1, 2, 3};
```

Massivda musbat elementlar soni va summasini hisoblash.

```
#include<stdio.h>

int main()
{
    int s = 0,k = 0;
    int x[] = {-1,2,5,-4,8,9};
    for(int i = 0; i<6; i++)
    {
        if (x[i]<= 0) continue;
        k++;
        s+ = x[i];
    };
    printf("%d\n",k);
    printf("%d\n",s);
    return 0;
};
```

Massivning eng katta, eng kichik elementi va o'rta qiymatini aniqlash:

```
#include<stdio.h>

int main()
{
    int i,j,n;
    float min,max,s = 0;
    float a,b,d,x[100];
    while(1)
    {
        printf("\n n = ");
```

```

scanf("%d",&n);
if ( n>0 && n<= 100 ) break;
printf("\n Hato 0<n<101 bo'lishi kerak");
}
printf("\n elementlar qiymatlarini kiriting:\n");
for (i = 0;i<n;i++)
{
printf("x[%d] = ",i);
scanf("%f",&x[i]);
}
max = x[0];
min = x[0];
for(i = 0;i<n;i++)
{
s+ = x[i];
if (max<x[i]) max = x[i];
if (min>x[i]) min = x[i];
};
s/ = n;
printf("\n max = %f",max);
printf("\n min = %f",min);
printf("\n o'rta qiymat = %f",s);
return 0;
};

```

Massivlarni navlarga ajratish. Navlarga ajratish - bu berilgan ko'plab obyektlarni biron-bir belgilangan tartibda qaytadan guruhlash jarayoni.

Massivlarning navlarga ajratilishi tez bajarilishiga ko'ra farqlanadi. Navlarga ajratishning $n \cdot n$ ta qiyoslashni talab qilgan oddiy usuli va $n \cdot \log(n)$ ta qiyoslashni talab qilgan tez usuli mavjud. Oddiy usullar navlarga ajratish tamoyillarini tushuntirishda qulay hisoblanadi, chunki sodda va kalta algoritmlarga ega.

Murakkablashtirilgan usullar kamroq sonli operatsiyalarni talab qiladi, biroq operatsiyalarning o'zi murakkabroq, shuning uchun uncha katta bo'lmagan massivlar uchun oddiy usullar ko'proq samara beradi.

Oddiy usullar uchta asosiy kategoriyaga bo'linadi:

- oddiy kiritish usuli bilan navlarga ajratish;
- oddiy ajratish usuli bilan navlarga ajratish;
- oddiy almashtirish usuli bilan navlarga ajratish.

Oddiy kiritish usuli bilan navlarga ajratish

Massiv elementlari avvaldan tayyor berilgan va dastlabki ketma-ketliklarga bo'linadi. $i = 2$ dan boshlab, har bir qadamda dastlabki ketma-ketlikdan i -nchi element chiqarib olinadi hamda tayyor ketma-ketlikning kerakli o'rniga kiritib qo'yiladi. Keyin i bittaga ko'payadi va h.k.

Kerakli joyni izlash jarayonida, ko'proq o'ngdan bitta pozitsiyadan tanlab olingan elementni uzatish amalga oshiriladi, ya'ni tanlab olingan element, $j = i - 1$ dan boshlab, navlarga ajratib bo'lingan qismning navbatdagi elementi bilan qiyoslanadi. Agar tanlab olingan element $a[i]$ dan katta bo'lsa, uni navlarga ajratish qismiga qo'shadilar, aks holda $a[j]$ bitta pozitsiyaga suriladi, tanlab olingan elementni esa navlarga ajratilgan ketma-ketlikning navbatdagi elementi bilan qiyoslaydilar. To'g'ri keladigan joyni qidirish jarayoni ikkita turlicha shart bilan tugallanadi:

agar $a[j] > a[i]$ elementi topilgan bo'lsa;

agar tayyor ketma-ketlikning chap uchiga bo'lsa.

int i, j, x;

for(i = 1; i < n; i++)

{

x = a[i]; // kiritib qo'yishimiz lozim bo'lgan elementni esda saqlab

qolamiz

j = i - 1;

while(x < a[j] && j >= 0) // to'g'ri keladigan joyni qidirish

}

a[j+1] = a[j]; // o'ngga surish

```

j--;
}
a[j+1] = x; //elementni kiritish
}

```

Oddiy tanlash usuli bilan navlarga ajratish

Massivning minimal elementi tanlanadi hamda massivning birinchi elementi bilan joy almashtiriladi. Keyin jarayon qolgan elementlar bilan takrorlanadi va h.k.

```

int i, min, n_min, j;
for(i = 0; i < n-1; i++)
{
min = a[i]; n_min = i; //minimal qiymatni qidirish
for(j = i + 1; j < n; j++)
if(a[j] < min){min = a[j]; n_min = j;}
a[n_min] = a[i]; //almashtirish
a[i] = min;
}

```

Oddiy almashtirish usuli bilan navlarga ajratish

Elementlar juftlari oxirgisidan boshlab qiyoslanadi va o'rin almashinadi. Natijada massivning eng kichik elementi uning eng chapki elementiga aylanadi. Jarayon massivning qolgan elementlari bilan davom yettiriladi.

```

for(int i = 1; i < n; i++)
for(int j = n - 1; j >= i; j--)
if(a[j] < a[j-1])
{int r = a[j]; a[j] = a[j-1]; a[j - 1] = r;}
}

```

Ko'p o'lchovli massivlar ta'rifi. Ikki o'lchovli massivlar matematikada matritsa yoki jadval tushunchasiga mos keladi. Jadvallarni inisializasiya qilish qoidasi, ikki o'lchovli massivning elementlari massivlardan iborat bo'lgan bir o'lchovli massiv ta'rifiga asoslangandir.

Misol uchun ikki qator va uch ustundan iborat bo'lgan haqiqiy turga tegishli d massiv boshlang'ich qiymatlari quyidagicha ko'rsatilishi mumkin:

```
float d[2][3] = {(1, -2.5, 10),(-5.3, 2, 14)};
```

Bu yozuv quyidagi qiymat berish operatorlariga mosdir:

```
d[0][0] = 1; d[0][1] = -2.5; d[0][2] = 10;
```

```
d[1][0] = -5.3; d[1][1] = 2; d[1][2] = 14;
```

Bu qiymatlarni bitta ro'yxat bilan hosil qilish mumkin:

```
float d[2][3] = {1, -2.5, 10, -5.3, 2, 14};
```

Initsializatsiya yordamida boshlang'ich qiymatlar aniqlanganda massivning hamma elementlariga qiymat berish shart emas.

Misol uchun: `int x[3][3] = {(1, -2, 3), (1, 2), (-4)}.`

Bu yozuv quyidagi qiymat berish operatorlariga mosdir:

```
x[0][0] = 1; x[0][1] = -2; x[0][2] = 3;
```

```
x[1][0] = -1; x[1][1] = 2; x[2][0] = -4;
```

Initsializatsiya yordamida boshlang'ich qiymatlar aniqlanganda massivning birinchi indeksi chegarasi ko'rsatilishi shart emas, lekin qolgan indekslar chegaralari ko'rsatilishi shart.

Misol uchun:

```
double x[][2] = {(1.1,1.5),(-1.6,2.5),(3,-4)}
```

Bu misolda avtomatik ravishda qatorlar soni uchga teng deb olinadi.

Quyidagi ko'radigan misolimizda jadval kiritilib har bir qatorning maksimal elementi aniqlanadi.

```
#include <stdio.h>
```

```
#include <stdlib.h>
```

```
int main(){
```

```
int a[4][3];
```

```
int max;
```

```
int i,j;
```

```
for(i = 0;i < 4; i++)
```

```
{
```

```

for(j = 0; j < 3; j++)
{
printf("a[%d][%d] = ",i,j);
scanf("%d",&a[i][j]);
}
printf("\n");
};
for(i = 0; i < 4; i++)
{
max = a[i][0];
for(j = 0; j < 3; j++)
if (max<a[i][j]) max = a[i][j];
printf("%d ",max);
}
system("pause");
return 0;
}

```

1-misol. Massiv elementlaridan musbatlarining soni va summasini hisoblash dasturi.

```

#include <iostream.h>
#include <conio.h>
void main ( )
{ int x[ ] = {1,2,56,78,-7,-45,34,12,9,-1};
int s=0, n=0; clrscr ( );
for (int i = 0; i < 10; i++)
{ if (x[i] < 0) continue;
s = s + x[i]; n++;}
cout << "s=" << s << "n=" << n << endl;
getch ( );
}

```

```

yoki
int x[10], s=0, i, n=0;
for (i = 0; i < 10; i++)
cin >> x[i];
for (i=0; i<10; i++)
{ if (x[i] < 0) continue;
s += x[i]; n++ ;
}

```

2-misol. 10ta elementdan iborat massivning eng katta, eng kichik elementlarini va ularning o'rtta qiymatini hisoblash dasturi.

```
# include <iostream.h>
# include <conio.h>
void main ( )
{
float x[] = { 1,2.23,5.6,-78,-7,-45.12,34.0,12,9,-1 };
float s, min, max; int i; clrscr ( );
min = x[0]; max=x[0];
for (i=0; i<10; i++)
{ if (min > x[i]) min = x[i];
  if (max < x[i]) max = x[i]; }
s = (min + max) / 2;
cout << "min="<<min<<endl;
cout <<"max="<<max<<endl;
cout <<"o'rta qiymat="<<s<<endl;
getch ( );
}
```

1-misol. 6ta elementdan iborat a va v massivlari berilgan. Ularni o'zaro qo'shib yangi s massivini hosil qiling.

```
# include <iostream.h>
# include <conio.h>
void main ( )
{ int a[6] = { 1,2,3,4,5,6},
b[6] = { 7,8,9,10,11,12}, c[6];
for (int i=0; i<6; i++)
{ c[i] = a[i] + b[i];
```

```

cout <<"c="<<c[i]<<endl; }

getch ( );

}

```

2-misol. Vektorlarning skalyar ko'paytmasini hisoblash dasturini tu'zing.

$$S = \sum_{i=1}^n x_i y_i ; n=5;$$

```

# include <iostream.h>
# include <conio.h>
void main ( )
{ int x[5]={1,2,3,4,5}, y[5]={6,7,8,9,10}, s=0;
for (int i=0; i < 5; i++)
s = s + x[i] * y[i];
cout << "s=" << s << endl;
getch ( ); }

```

3-misol. Bazaviy toifasi haqiqiy bo'lgan 10ta elementli A massivi berilgan. Juft indeksli elementlardan alohida, toq indeksli elementlardan alohida massiv hosil qiling.

```

# include <iostream.h>
# include <conio.h>
void main ( )
{ float a[10], b[5], c[5];
for (int i=0; i<10; i++)
cin >> a[i];
for (int i=0; i<5; i++)
{ c[i] = a[2*i +1];
b[i] = a[2*i];
}
}

```

```

cout << "c=" << c[i];
cout << "b=" << b[i] <<endl; }
getch ( );
}

```

4-misol. A va V massivlari berilgan. Yangi S massivini quyidagicha hosil qiling: A massivning elementlari yangi massivning toq elementlari, V massivning elementlari esa yangi massivning juft elementlarini tashkil etadi.

```

# include <iostream.h>
# include <conio.h>
void main ( )
{ float a[5], b[5], c[10]; int i;
for ( i=0; i<5; i++)
cin >> a[i] >> b[i];
for ( i=0; i< 5; i++)
{ c[2*i+1] = b[i];
  c[2*i] = a[i]; }
for (i = 0; i < 10; i++)
  cout << "c[" << i << "]=" << c[i] <<endl ;
getch ( );
}

```

Nazorat uchun savollar

1. Massivlar haqida tushuncha?
2. Bir o'lchamli massivlarni tashkil etish, o'zgartirish?
3. Ko'p o'lchamli massivlarni tashkil etish, o'zgartirish?
4. Massivlarni saralash?

BAJARISH UCHUN TOPSHIRIQLAR

Jadval 7.1

№	Bir o'lchovli massivlar (ixtiyoriy o'lchamli massiv)
1	$U(K)$ massivi berilgan. $[a,b]$ intervaliga tegishli bo'lgan elementlar sonini aniqlang va ekranga chop eting.
2	$R(N)$ massivi berilgan. D massivga uning maksimal va minimal elementidan tashqari bo'lgan elementlarni yozing
3	$X(N)$ massivining o'rta arifmetik va o'rta geometrik qiymatlarini aniqlang. X massivining o'rta arifmetik qiymatidan katta va o'rta geometrik qiymatdan kichik bo'lgan elementlar sonini aniqlang va chop eting.
4	$U(K)$ massivining musbat, toq indeksli elementlarining yig'indisini toping.
5	$Z(K)$ massivining manfiy, juft indeksli elementlarining ko'paytmasini toping.
6	$H(N)$ massivining o'rta arifmetik qiymatidan kichik bo'lan elementlarining ko'paytmasini toping.
7	$H(N)$ massivining eng katta va eng kichik elementlarini toping va massivda ulardan qaysi biri avval joylashganini aniqlang.
8	Berilgan ikkita $X=(X_1, X_2, \dots, X_n)$ va $Y=(Y_1, Y_2, \dots, Y_n)$ vektorning skalyar ko'paytmasini toping.
9	$X(N)$ massiv berilgan. X massiv elementlarini kamayib borish tartibida joylang.
10	$X(N)$ massivining musbat elementlaridan Y massivini tashkil eting, so'ngra Y massiv elementlarini o'sib borish tartibida yozing
11	$X(N)$ massivining $X_i > M$, shartini qanoatlantiruvchi elementlar yig'indisini hisoblang. (bunda $M = \max X - \min X$).
12	$Y(K)$ massivdan X massivga manfiy elementlarni ko'chirib yozing. X massivdagi eng kichik elementni birinchi element bilan o'rnini almashtiring.
13	$X(N)$ va $Y(K)$ ikkita massiv berilgan. X va Y massivlari-ning musbat elementlaridan Z massivni tashkil eting. Z massivining maksimal elementini va uning indeksini aniqlang.
14	$X(N)$ massivi berilgan. Uning elementlarini teskari tartibda joylashtiring.

	Massivning toq indeksli element-larining ko'paytmasini hisoblang.
15	$X(N)$ massiv berilgan. Uning juft indeksli elementlarni Z massivga ko'chirib yozing. Z massivdagi maksimal va minimal elementlarining o'rnini o'zaro almashtiring.
16	$X(K)$ massiv berilgan. X massivdagi manfiy elementlarni Z massivga ko'chirib yozing. Hosil bo'lgan Z massivni kamayib borish tartibida joylang.
17	$Z(N)$ massivi berilgan. Y massivni shunday tashkil eting-ki, Y massivida avval musbat, keyin manfiy, va nul elementlari joylashsin.
18	Z massivdagi manfiy elementlar kupaytmasini, hamda yig'indisini hisoblang.
19	$Z(K)$ massivining maksimal va minimal elementidan boshqa barcha elementlaridan X massivini xosil qiling. X massivining o'rta geometrik qiymatidan kichik bo'lgan yele-mentlari sonini toping.
20	$X(N)$ massivdagi xamma musbat elementlaridan Z massivini xosil qiling va Z massiv elementlarini kamayib borish tartibida joylang.
21	$X(K)$ massivlari berilgan. X massivdagi barcha musbat elementlari indeksidan L massivini xosil qiling.
22	$X(N)$ i $Y(N)$ massivlarining manfiy elementlaridan D massivini xosil qiling. D massiv elementlarini o'sib borish tartibida joylang. 3 ta minimal elementlarni chop eting.
23	$X(N)$ va $Y(K)$ massivlari berilgan. D massivini shunday tashkil etingki, unda X massivining 3ta va U massivining 4 ta eng katta elementlari yozilgan bo'lsin. D massivi element-larining kupaytmasi xisoblang.
24	$X(N)$ massivning modul bo'yicha eng katta elementini toping va uni birinchi elementi bilan o'rnini almashtiring.
25	$X(N)$ va $Y(N)$ massivning musbat elementlarining yig'in-disini va manfiy elementlarining kupaytmasi xisoblang.
26	$X(N)$ va $Y(N)$ massivlari berilgan. $A_i = X_i + Y_i$ formulasi yordamida massiv tashkil eting. A massivning o'rta arif -metik qiymatidan kichik bo'lgan

	elementlar sonini ,xamda eng katta va eng kichik elementlar orasidagi farqni xisoblang.
27	$X(N)$ biror to'g'ri chiziqqa tegishli bo'lgan nuqtalar koordinatalar to'plamidan iborat massiv bo'lsin. Shu massivga tegishli bo'lgan , qaysi ikki nuqta orasidagi masofa eng katta yekanligini aniqlang.
28	$X(N)$ massivi berilgan. $X(N)$ massivi o'rta arifmetik qiymatini aniqlang va massiv birinchi elementining o'rni bilan almashtiring.
29	$X(N)$ massivi berilgan. Ularni kamayib borish tartibida $Y(N)$ massivga joylang. $X(N), Y(N)$ massivlarning modul bo'yicha eng katta elementining indeksini aniqlang .
30	$Z(N)$ massivi berilgan. Uning elementlarining moduli bo'yicha o'sib borish tartibida joylang.

Ikki o'lchamli massivlarga doir variantlar

Jadval 7.2

1. Matritsa elementlari ichidan eng katta qiymatini va uning indekslarini topuvchi dastur tuzing (acm.tuit.uz, N003).

Kiruvchi ma'lumotlar: Birinchi satrda matritsaning satrlar va ustunlar soni n va m ($0 \leq n, m \leq 10$) berilgan. Keyingi n ta satrlarda m tadan massiv elementlari berilgan ($-10^{38} \leq a_{ij} \leq 10^{38}$).

Chiquvchi ma'lumotlar: Matritsaning max qiymatini va uning indekslarini chiqaring. Haqiqiy sonlarni 10^{-2} aniqlikda chiqaring.

Kiritishga misol	Chiqarishga misol
3 3	9.00 2 2
1 2 3	
4 5 6	
7 8 9	

4 3	95511529.00 3 0
91259790 71414149 54400441	
-4471842 35270592 17346684	
77615570 16486050 52824525	
95511529 62664009 21913614	

2. Kvadrat matritsani uning o'ziga bo'lib yangi matritsa hosil qilivchi dastur tuzing (acm.tuit.uz, N006).

Kiruvchi ma'lumotlar: Birinchi satrda kvadrat matritsaning o'lchami ($0 \leq n \leq 10$) berilgan. Keyingi satrda matritsa elementlari kiritilgan ($-10^{38} \leq a_{ij} \leq 10^{38}$).

Chiquvchi ma'lumotlar: Yangi matritsa elementlarini matritsa shaklida chiqaring. Haqiqiy sonlarni 10^{-2} aniqlikda chiqaring. Agarda bunday matritsani topib bo'lmasa No solution chiqaring.

Kiritishga misol	Chiqarishga misol
3	0.07 0.13 0.20
1 2 3	0.27 0.33 0.40
4 5 6	0.47 0.53 0.60
7 8 9	

3. Ixtiyoriy matritsaning min elementini aniqlab, so'ngra shu element turgan qatorni olib tashlovchi dastur tuzing (acm.tuit.uz, N005).

Kiruvchi ma'lumotlar: Birinchi satrda matritsaning qatorlar va ustunlar soni n va m ($0 \leq n, m \leq 100$) berilgan. Keyingi satrda matritsa elementlari kiritilgan ($-10^{38} \leq a_{ij} \leq 10^{38}$).

Chiquvchi ma'lumotlar: Matritsaning min elementini turgan qatorni olib tashlab, qolgan elementlarini matritsa shaklida chiqaring.

Kiritishga misol	Chiqarishga misol
3 3 1 2 3 4 5 6 7 8 9	4 5 6 7 8 9

4. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring. Bosh diogonal va unga parallel bo'lgan dioganallarining eng katta elementlarini chiqaring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3 1 2 3 4 5 6 7 8 9	8 9 6
5 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	22 23 24 25 20 15 10

5. Kvadrat matritsa berilgan. Berilgan matritsaga teskari bo'lgan matritsani topuvchi dastur tuzing. Agarda berilgan matritsa uchun teskari matritsa mavjud bo'lmasa False chiqaring (acm.tuit.uz, M008).

Kiruvchi ma'lumotlar: Kvadrat matritsaning parametri n berilgan ($1 \leq n \leq 100$). Keyingi n ta satrda har birida n tadan butun sonlardan iborat bo'lgan matritsaning elementlari probel orqali berilgan.

Chiquvchi ma'lumotlar: Berilgan kvadrat matritsaga teskari bo'lgan matritsani elementlarini n ta satr va n ta ustunda probel orqali chiqaring. Teskari matritsa mavjud bo'lmasa False chiqaring. Haqiqiy sonlarni 10^{-1} aniqlikda chiqaring.

Kiritishga misol	Chiqarishga misol
2 1 2 3 4	-2 1 1.5 -0.5
2 1 3 1 3	False

6. $N \times N$ o'lchovli matritsa berilgan. Shu matritsa elementlaridan bir o'lchovli massiv hosil qiluvchi dastur tuzing (acm.tuit.uz, M007).

Kiruvchi ma'lumotlar: Birinchi satrda matritsaning o'lchovi n ($1 \leq n \leq 10$) berilgan. Ikkinchi satrda massivning n ta a_i elementi ($-10^6 \leq a_i \leq 10^6$) probel orqali berilgan.

Chiquvchi ma'lumotlar: Masala shartini qanoatlantiruvchi massiv elementlarini chiqaring.

Kiritishga misol	Chiqarishga misol
3 1 2 3 5 7 9 2 9 5	1 2 3 5 7 9 2 9 5

7. Butun $N \times M$ ($2 \leq N, M < 100$) matritsa berilgan. Ilon bo'yicha to'ldirish (acm.tuit.uz, L2132).

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani shart bo'yicha to'ldiring.

Sonlarni probel bilan chiqaring.

Kiritishga misol	Chiqarishga misol
2 2	0 1 3 2
3 3	0 1 2 5 4 3 6 7 8

8. Butun $N \times M$ ($2 \leq N, M < 25$) matritsa berilgan. Uni quyidagi qoida bo'yicha to'ldiring: 0 satr va 0 ustundagi barcha elementlar 1 ga teng. Qolgan barcha elementlari esa elementning chap tarafida va ustida turgan elementlar yig'indisiga teng (acm.tuit.uz, L2131).

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel bilan berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani paskal uchburchagi shaklida to'ldiring.

Kiritishga misol	Chiqarishga misol
2 2	1 1 1 2
2 3	1 1 1 1 2 3

9. Butun $N \times N$ ($2 \leq N \leq 100$) matritsa berilgan. Uni quyidagicha to'ldiring (acm.tuit.uz, L2129):

- chap diagonalini bir bilan;
- chap diagonaldan tepasini 0 bilan;

- chap diagonaldan quyisini 2 bilan.

Kiruvchi ma'lumotlar: Birinchi satrda N soni probel berilgan.

Chiquvchi ma'lumotlar: N x N o'lchamli matritsani shart bo'yicha to'ldiring.

Sonlarni probel bilan chiqaring.

Kiritishga misol	Chiqarishga misol
2	0 1 1 2
3	0 0 1 0 1 2 1 2 2

10. Butun N x N ($2 < N < 100$) matritsa berilgan. Matritsani bir va nol bilan spiral shaklda to'ldiring. Birni chap yuqori burchakdan soat strelkasi bo'yicha kiriting (acm.tuit.uz, L2126):

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: N x N o'lchamli matritsani spiral shaklda to'ldiring.

Kiritishga misol	Chiqarishga misol
3	111 001 111
5	11111 00001 11101 10001 11111

11. Butun N x M ($2 < N, M < 100$) matritsa berilgan. Uni dioganal bo'yicha to'ldiring. Sonlar bitta probel bilan ajratilgan (acm.tuit.uz, L2127):

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel bilan berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani diogonal shaklida to'ldiring.

Kiritishga misol	Chiqarishga misol
2 2	0 1 2 3
2 3	0 1 3 2 4 5

12. Butun $N \times M$ ($2 \leq N, M < 100$) matritsa berilgan. Uni elementlarini faqat chet taraflardagi elementlarini bitta probel bilan ajratgan holda 1 bilan qolgan elementlarini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel bilan berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
2 2	1 1 1 1
3 3	1 1 1 1 0 1 1 1 1

13. Butun $N \times M$ ($2 \leq N, M < 100$) matritsa berilgan. Uning o'ng va chap diagonallarini 1 bilan qolgan elementlarini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
2 2	1 1 1 1
3 3	1 0 1

	0 1 0
	1 0 1

14. Butun $N \times M$ ($5 \leq N, M < 100$) matritsa berilgan. Uning o'ng va chap diagonallarini ikkita qo'shni 1 bilan qolgan elementlarini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
5 5	1 1 0 1 1 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 1 1 0 1 1
7 7	1 1 0 0 0 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 0 0 0 1 1 1 0 0 0 1 1 1 1 1 0 1 1 1 0 1 1 1 1 1 0 0 0 1 1

15. Butun $N \times M$ ($2 \leq N, M < 100$) matritsa berilgan. Uning o'ng va chap diagonallarini tepa va past qismini 1 bilan qolgan elementlarini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel bilan berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3 3	0 1 0 0 0 0 0 1 0
5 5	0 1 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 0

16. Butun $N \times M$ ($2 \leq N, M < 100$) matritsa berilgan. Uning o'ng va chap diagonallarini o'ng va chap qismini 1 bilan qolgan elementlarini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N va M soni probel bilan berilgan.

Chiquvchi ma'lumotlar: $N \times M$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3 3	0 0 0 1 0 1 0 0 0
5 5	0 0 0 0 0 1 0 0 0 1 1 1 0 1 1 1 0 0 0 1 0 0 0 0 0

17. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsaning ustun va satrini o'rtalarini tutashtirishdan hosil bo'lgan kvadratni 1 bilan qolgan qismini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: N x N o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	0 1 0 1 1 1 0 1 0
5	0 0 1 0 0 0 1 1 1 0 1 1 1 1 1 0 1 1 1 0 0 0 1 0 0

18. Butun N x N ($3 \leq N < 100$) matritsa berilgan. Matritsaning bosh diagonalidan yuqori elementlarini 1 bilan qolgan qismini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: N x N o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	1 1 1 0 1 1 0 0 1
5	1 1 1 1 1 0 1 1 1 1 0 0 1 1 1 0 0 0 1 1 0 0 0 0 1

19. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsaning bosh diagonalidan pastgi elementlarini 1 bilan qolgan qismini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	1 0 0 1 1 0 1 1 1
5	1 0 0 0 0 1 1 0 0 0 1 1 1 0 0 1 1 1 1 0 1 1 1 1 1

20. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsaning chap diagonalidan yuqori elementlarini 1 bilan qolgan qismini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	1 1 1 1 1 0 1 0 0
5	1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 0 0 0 1 0 0 0 0

21. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsaning chap diagonalidan pastki elementlarini 1 bilan qolgan qismini esa 0 bilan to'ldiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	0 0 1 0 1 1 1 1 1
5	0 0 0 0 1 0 0 0 1 1 0 0 1 1 1 0 1 1 1 1 1 1 1 1 1

22. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uni 90^0 gradusga buring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	7 4 1
1 2 3	8 5 2
4 5 6	9 6 3
7 8 9	

23. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uni -90^0 gradusga buring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	3 6 9
1 2 3	2 5 8
4 5 6	1 4 7
7 8 9	

24. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uni 180^0 gradusga buring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	9 8 7
1 2 3	6 5 4
4 5 6	3 2 1
7 8 9	

25. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uni -180^0 gradusga buring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	9 8 7
1 2 3	6 5 4
4 5 6	3 2 1
7 8 9	

26. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uni elementlarini chap diogonal bo'yicha simmetrik almashtiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	9 6 3
1 2 3	8 5 2
4 5 6	7 4 1
7 8 9	

27. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uni elementlarini bosh diogonal bo'yicha simmetrik almashtiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	1 4 7
1 2 3	2 5 8
4 5 6	3 6 9
7 8 9	

28. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uning elementlarini eng markaziy satri bo'yicha simmetrik almashtiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	7 8 9
1 2 3	2 5 8
4 5 6	1 2 3
7 8 9	

29. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring va uning elementlarini eng markaziy ustuni bo'yicha simmetrik almashtiring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3	3 2 1
1 2 3	6 5 4
4 5 6	9 8 7
7 8 9	
5	5 4 3 2 1
1 2 3 4 5	10 9 8 7 6
6 7 8 9 10	15 14 13 12 11
11 12 13 14 15	20 19 18 17 16
16 17 18 19 20	25 24 23 22 21
21 22 23 24 25	

30. Butun $N \times N$ ($3 \leq N < 100$) matritsa berilgan. Matritsani 1 dan $N \times N$ gacha bo'lgan sonlar bilan to'ldiring. Bosh dioganali va unga parallel bo'lgan dioganallarining eng katta (eng kichik) elementlarini chiqaring.

Kiruvchi ma'lumotlar: Birinchi satrda N soni berilgan.

Chiquvchi ma'lumotlar: $N \times N$ o'lchamli matritsani masala sharti bo'yicha chiqaring.

Kiritishga misol	Chiqarishga misol
3 1 2 3 4 5 6 7 8 9	4 1 2
5 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25	16 11 6 1 2 3 4

Jadval 7.3

1.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa ustunlarida turuvchi elementlar ichidan eng kattalari yig'indisi topilsin.
2.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa satrlarida turuvchi elementlar ichidan eng kichiklari ko'paytmasi topilsin.
3.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa ustunlarida turuvchi elementlar ichidan eng kichiklarining o'rta arifmetigi topilsin.
4.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa satrlarida turuvchi elementlar ichidan eng kattalarining o'rta geometrigi topilsin.
5.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa elementlarining o'rtacha qiymati topilsin. Massivda o'rtacha qiymatdan kichik va katta elementlari soni topilsin.
6.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa ustunlarida 4-ga karrali element bor yo'qligini aniqlansin. Agar bor bo'lsa ustunga mos $S(n)$ massiv elementiga 1 qiymat aks holda 0 qiymat berilsin, ya'ni $S(n)$ massiv elementlari qiymati topilsin.

7.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisa ustunlaridagi 2-ga karralilarining yig'indisini mos $B(n)$ massiv elementiga joylashtirilsin.
8.	Haqiqiy elementlardan tashkil topgan kvadrat matrisa $A=\{a_{ij}\}$ berilgan. Nol va birlardan tashkil topgan b_1, b_2, \dots, b_n ketma-ketlik tuzilsin, unda $b_i=1$ agar matrisa satrlarida hech bo'lmaganda 1 bilan 10 orasida yotuvchi bitta element mavjud bo'lsa.
9.	Haqiqiy elementlardan tashkil topgan kvadrat matrisa $A=\{a_{ij}\}$ berilgan. Nol va birlardan tashkil topgan b_1, b_2, \dots, b_n ketma-ketlik tuzilsin, unda $b_i=1$ agar matrisa ustunlarida hech bo'lmaganda bitta 2-ga karrali element mavjud bo'lsa.
10.	Ikkita $A(n)$ va $B(m)$ massivlari berilgan. Bu massivlar elementlari ichida nechtadan manfiy va musbat elementlar borligi topilsin.
11.	$A=\{a_{ij}\}$ matrisa asosiy diagonalidagi elementlar o'rta arifmetigi topilsin. Bu yerda $a_{ij}=(5^i-3^j)/n$; $ij=1,2,\dots,n$.
12.	$A=\{a_{ij}\}$ matrisa ustun elementlari orasida 4-ga karralilari soni topilsin. Bu yerda $a_{ij}=4^i$ agar $i=j$ bo'lsa, $a_{ij}=2^{i+j}$ agar $i \neq j$ bo'lsa; $ij=1,2,\dots,n$.
13.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisaning juft nomerli ustun elementlari orasidan 2 va 4-ga karralilari soni topilsin.
14.	$A=\{a_{ij}\}$ matrisa satr elementlari orasida 2-ga karralilari soni topilsin. Bu yerda $a_{ij}=2^{i+j}$ agar $i+j>3$ bo'lsa, $a_{ij}=5$ agar $i+j \leq 3$ bo'lsa; $i=1,2,\dots,m$; $j=1,2,\dots,n$.
15.	$A=\{a_{ij}\}$ matrisa berilgan. Matrisaning toq nomerli satr elementlari orasida 3-ga karralilarning o'rta arifmetigi topilsin.
16.	X son va $A=\{a_{ij}\}$ matrisa berilgan. Matrisa elementlari ichida X sonidan kattalari soni topilsin va shu elementlar kvadratga oshirilsin.
17.	X son va $A=\{a_{ij}\}$ matrisa berilgan. Matrisa elementlari ichida X sonidan kattalari soni topilsin va shu elementlar ichida X^2 dan katta element mavjudligi aniqlansin.
18.	$A=\{a_{ij}\}$ matrisa berilgan bo'lsa quyidagilarni hisoblang: $x_i=\max\{a_{ij}\}$; $y_j=\min\{a_{ij}\}$.
19.	$A=\{a_{ij}\}$ matrisa berilgan bo'lsa quyidagilarni hisoblang:

	$x_i = \max\{a_{ij}\}; y = \min\{x_i\}; z = \max\{x_i\}.$
20.	$A = \{a_{ij}\}$ matrisa berilgan. Matrisa bosh diagonali va 1-chi ustun elementlari o'rtta arifmetigi topilsin va ular orasidan kattasi aniqlansin.
21.	$A = \{a_{ij}\}$ matrisa berilgan. Matrisa bosh diagonali va n-chi satr elementlari yig'indisi topilsin va ular orasidan kattasi aniqlansin.
22.	$A = \{a_{ij}\}$ va $B = \{b_{ij}\}$ matrisalari berilgan. Matrisa elementlari o'rtta arifmetigi hisoblansin va ular kattasi aniqlansin.
23.	$A = \{a_{ij}\}$ va $B = \{b_{ij}\}$ matrisalari berilgan. Bu matrisa elementlari eng kichigini topib, ulardan kattasi aniqlansin.
24.	Elementlari butun $A = \{a_{ij}\}$ matrisa berilgan. Matrisa elementlarining 2-ga karralilari va ularning sonlari aniqlansin.
25.	Elementlari butun $X(n)$ vektori berilgan. Vektorning juft qiymatli elementlarini oldin, toq qiymatli elementlarini keyin $Y(n)$ vektoriga joylashtirilsin.
26.	Elementlari butun $X(n)$ vektori berilgan. Vektorning manfiy qiymatli elementlarini oldin, musbat qiymatli elementlarini keyin $Y(n)$ vektoriga joylashtirilsin.
27.	$X(n)$ vektori berilgan. Vektorning elementlarini $Y(n)$ vektoriga teskari joylashtirilsin.
28.	$X(n)$ vektori va z soni berilgan. Vektorning elementlarini $Y(n)$ vektoriga oldin z sonidan kichiklari va keyin kattalari joylashtirilsin.
29.	Butun sonli $X(n)$ vektori berilgan. Vektorning eng katta elementining eng kichik umumiy bo'luvchisi topilsin.
30.	Butun sonli $X(n)$ vektori berilgan. Vektorning eng katta elementidan bitta oldingi eng katta elementini topib, uning eng katta umumiy bo'luvchisi topilsin.

8-Laboratoriya ishi. C++da funksiyalarni tashkil etish.

Ishning maqsadi

1. C++ da funksiyalarni tashkil etish bilan tanishish.
2. Funksiyalarda qiymatlarni uzatish.
3. Funksiya- protseduralarni tashkil etish.
4. Funksiyalarga qiymatlarni adres orqali uzatish.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarning dasturini yozadi.
2. Har bir vazifani bajarishda foydalanilgan o'zgaruvchi va funksiyalar haqida hisobotda ko'rsatish lozim.
3. Hisobot shaklida oldin vazifa, uni bajarishda foydalanilgan funksiyalar, dastur matni va bajarishdan hosil bo'lgan natijani keltirish lozim.
4. 8-laboratoriya ishida berilgan barcha vazifalarni bitta hisobot shaklida topshiring.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. Laboratoriya ishidagi topshiriq raqami
3. Topshiriqni bajarishda foydalanilgan funksiyalar tavsifi.
4. Topshiriq kodi.
5. Topshiriq natijasi.
6. Dasturning elektron versiyasi (Albatta bu dasturlar kompyuterda o'qituvchiga ko'rsatiladi. O'qituvchini talab va takliflariga ko'ra dastur to'g'rilanadi va so'ngra hisobot tayyorlanadi.).

NAZARIY QISM

Funksiyalarni ta'riflash va ularga murojaat qilish. Funksiya ta'rifida funksiya nomi, turi va formal parametrlar ro'yxati ko'rsatiladi. Formal parametrlar nomlaridan tashqari turlari ham ko'rsatilishi shart. Formal parametrlar ro'yxati funksiya signaturasi deb ham ataladi.

Funksiya ta'rifi umumiy ko'rinishi quyidagichadir:

Funksiya turi funksiya nomi(formal_parametrlar_ta'rifi)

Formal parametrlarga ta'rif berilganda ularning boshlang'ich qiymatlari ham ko'rsatilishi mumkin.

Funksiya qaytaruvchi ifoda qiymati funksiya tanasida **return** <ifoda> ; operatori orqali ko'rsatiladi.

Misol:

float min(float a, float b)

{

if (a<b) return a;

return b;

}

Funksiyaga murojaat qilish quyidagicha amalga oshiriladi:

Funksiya nomi (haqiqiy parametrlar ro'yxati)

Haqiqiy parametr ifoda ham bo'lishi mumkin. Haqiqiy parametrlar qiymati hisoblanib mos formal parametrlar o'rnida ishlatiladi.

Misol uchun yuqoridagi funksiya quyidagicha murojaat qilish mumkin:

int x = 5,y = 6,z; z = min(x,y) yoki **int z = min(5,6)** yoki **int x = 5; int z = min(x,6)**

Funksiyaga murojaat qilinganda haqiqiy parametrlar turlari formal parametrlar turlariga mos kelmasligi mumkin. Bu holda avtomatik ravishda turlarni keltirish bajariladi.

Funksiya qiymat qaytarmasa turi **void** deb ko'rsatiladi.

Misol uchun:

```
void print()  
{  
printf("\n Salom!");  
};
```

Bu funksiyaga print() shaklida murojaat qilish ekranga Salom! yozilishiga olib keladi.

Qiymat qaytarmaydigan funksiya tanasida return operatori ishlatilishi mumkin.

Bu operator funksiyadan chiqishni bildiradi. Masalan:

```
void print()  
{  
printf("\n Salom!");  
return;  
printf("\n Dunyo!");  
}
```

Bu funksiyaga print() shaklida murojaat qilish ekranga Salom! yozilishiga olib keladi, lekin **Dunyo!** so'zi yozilmay qoladi.

Qiymat qaytarmaydigan funksiya formal parametrlarga ega bo'lishi mumkin.

Masalan:

```
#include <stdio.h>  
void print_baho(int baho)  
{  
switch(baho)  
{  
case 2:printf("\n yomon");break;  
case 3:printf("\n o'rta");break;  
case 4:printf("\n yaxshi");break;  
case 5:printf("\n alo");break;  
default:printf("\n noto'ri kiritilgan");  
};
```

```

};
int main()
{
int a;
scanf("%d",&a);
print_baho(5);
return 0;
};

```

Funksiya prototipi. Agar programmada funksiya ta’rifi murojaatdan keyin berilsa, yoki funksiya boshqa faylda joylashgan bo’lsa, murojaatdan oldin shu funksiyaning prototipi joylashgan bo’lishi kerak. Prototip funksiya nomi va formal parametrlar turlaridan iborat bo’ladi. Formal parametrlar nomlarini berish shart emas.

Misol uchun $y = \min(a,b)+2*\max(c,d)$ ifodani hisoblashni ko’ramiz:

```

#include <stdio.h>
int max(int a,int b)
{
if (a<b) return b;else return a;
}
int main()
{
int a,b,c,d,y;
int min(int,int);
scanf("%d%d%d%d",&a,&b,&c,&d);
y = min(a,b)+2*max(c,d);
printf("\n %d",y);
return 0;
};
int min(int a,int b)
{

```

```
if (a<b) return a;  
else return b;  
}
```

Funksiyaga parametrlar uzatish. Funksiyaga parametrlar qiymat bo'yicha uzatiladi va quyidagi bosqichlardan iborat bo'ladi:

1. Funksiya bajarishga tayyorlanganda formal parametrlar uchun xotiradan joy ajratiladi, ya'ni formal parametrlar funksiyalarning ichki parametrlariga aylantiriladi. Agar parametr turi float bo'lsa double turidagi obyektlar hosil bo'ladi, char va short int bo'lsa int turidagi obyektlar yaratiladi.

2. Haqiqiy parametrlar sifatida ishlatilgan ifodalar qiymatlari hisoblanadi.

3. Haqiqiy parametrlar ifodalar qiymatlari formal parametrlar uchun ajratilgan xotira qismlariga yoziladi. Bu jarayonda float turi double turiga, char va short int turlari int turiga keltiriladi.

4. Funksiya tanasi ichki obyektlar – parametrlar yordamida bajariladi va qiymat chaqirilgan joyga qaytariladi.

5. Haqiqiy parametrlar qiymatlariga funksiya hech qanday ta'sir o'tkazmaydi.

6. Funksiyadan chiqishda formal parametrlar uchun ajratilgan xotira qismlari bo'shatiladi.

C tilida chaqirilgan funksiya chaqiruvchi funksiyadagi o'zgaruvchi qiymatini o'zgartira olmaydi. U faqat o'zining vaqtinchalik nusxasini o'zgartirishi mumkin xolos.

Qiymat bo'yicha chaqirish qulaylik tug'diradi. Chunki funksiyalarda kamroq o'zgaruvchilarni ishlatishga imkon beradi. Misol uchun shu xususiyatni aks ettiruvchi POWYeR funksiyasi variantini keltiramiz:

```
int powyer(int x, int n)  
{  
int p;  
for (p = 1; n > 0; --n)  
p = p * x;
```

```
return (p);
```

```
}
```

Argument *n* vaqtinchalik o'zgaruvchi sifatida ishlatiladi. Undan to qiymati 0 bo'lmaguncha bir ayriladi. Parametr *n* funksiya ichida o'zgarishi funksiyaga murojaat qilingan boshlang'ich qiymatiga ta'sir qilmaydi.

Masalan: sonning kubini hisoblash uchun funksiya tashkil eting va undan foydalaning.

```
# include <iostream.h>
```

```
# include <conio.h>
```

```
void main ( )
```

```
{ int k, n, kw (int n); // kw - funksiya nomi (ixtiyoriy)
```

```
cin>>n; // n - berilayotgan son
```

```
k=kw(n); // kw funksiyasiga murojaat qilinyapti
```

```
cout << "k="<<k<<endl;
```

```
getch( );
```

```
}
```

```
int kw (int a) // funksiya aniqlanayapti. Bu yerda a rasmiy parametr
```

```
{ int c; // lokal o'zgaruvchi
```

```
c=a*a*a; // hisoblash
```

```
return c; } // funksiyaga natijani kaytarish
```

Yuqoridagi *s* lokal o'zgaruvchisini ishlatmasdan, to'g'ridan-to'g'ri

```
return a*a*a; deb yozsa ham bo'ladi.
```

Bu yerda funksiya bosh funksiyadan keyin aniqlandi, shuning uchun uni bosh funksiya ichida e'lon qildik. Dasturni yana quyidagicha yozsa ham bo'ladi:

```
# include <iostream.h>
```

```
# include <conio.h>
```

```
int kw (int a)
```

```
{ return a*a*a; }
```

```
void main ( )
```

```
{ int k, n ;
```

```

cin>>n;
k=kw(n);
cout << "k="<<k<<endl;
getch( );
}

```

2-misol. Ikkita sondan eng kattasini topish uchun funksiya tashkil qiling va undan foydalaning.

```

#include <iostream.h>
#include <conio.h>
void main( )
{ float a=7, b=9, c, max(float , float );
c = max(a, b);
cout << "c="<<c<<endl;
getch( );
}
float max ( float x, float y)
{ if (x > y) return x; else return y; }

```

Funksiyaga yana quyidagicha ham murojaat qilish mumkin:

```

c = max( 7.23, 9.145);
c = max( a, 9.145);

```

3-misol. Uchburchak uchlarining koordinatalari berilgan. Shu koordinatalar yordamida uchburchak qursa bo'ladimi? Agar mumkin bo'lsa shu uchburchakning yuzini hisoblash dasturini tuzing.

Demak, berilgan koordinatalari yordamida uchburchak tomonini ko'rish funksiyasini, shu tamonlar asosida uchburchak ko'rish mumkinmi yoki yo'qligini va uning yuzini hisoblash funksiyalarini tuzing.

```

#include <iostream.h>
#include <math.h>
#include <conio.h>
// uchburchak tomonini topish funksiyasi

```



```

float line (float x1, float x2, float y1, float y2)
{ (float) p = sqrt ((x1-x2)*(x1-x2)+ (y1-y2)*(y1-y2));
  return p; }

// uchburchak ko'rib bo'ladimi? funksiyasi
int uch ( float a, float b, float c)
{ if ( a+b>c && b+c>a && c+a>b ) return 1;
  else return 0; }

// uchburchakning yuzini topish funksiyasi
float s (float a, float b, float c)
{ float p, s ;
  p = ( a + b + c ) / 2; s = sqrt (p*(p-a)*(p-b)*(p-c));
  return s; }

void main ( )
{ float x1, x2, x3, y1, y2, y3, p1, p2, p3; clrscr ( );
  cin >> x1>> x2>> x3>> y1>> y2>> y3;
  p1 = line (x1, x2, y1, y2);
  p2 = line (x1, x3, y1, y3);
  p3 = line (x2, x3, y2, y3);
  t = uch (p1, p2, p3);
  if ( t == 1)
  { yuza = s ( p1, p2, p3); cout << "yuza = " << yuza << endl;
  else cout << "uchburchak qurib bo'lmaydi !!!" << endl;
  } getch ( ); }

```

Bir funksiya ichida boshqa funksiya aniqlanishi mumkin emas, lekin funksiya ichida o'zini-o'zi chaqirishi mumkin, bunday holatni rekursiya holati deyiladi. Rekursiya 2 xil bo'ladi: to'g'ri rekursiya va bilvosita rekursiya. Agar funksiya o'zini-o'zi chaqirsa, bu to'g'ri rekursiya deyiladi. To'g'ri rekursiyada funksiyaning nusxasi chaqiriladi. Agarda funksiya boshqa bir funksiyaning chaqirsa va u funksiya o'z navbatida 1-chisini chaqirsa, u holda bilvosita rekursiya deyiladi. Rekursiya 2 xil natija bilan yakunlanadi: biror natija qaytaradi yoki hech qachon tugallanmaydi va

xatolik yuz beradi. Bunday holatlarda rekursiv funksiyalar uchun rekursiyani to'xtatish shartini berish zarur, chunki rekursiyada xotira yetishmasligi havfi bor.

4-misol. $F = n!$ ni hisoblash uchun funksiya tashkil eting va undan foydalaning.

```
# include <iostream.h>
# include <conio.h>
void main( )
{ int n, f, fac(int);
cout <<"sonni kiriting:"; cin >> n;
f = fac(n); cout <<"sonning faktoriali="<<f<<endl;
getch( );
}
int fac(int i)
{ return i <=1 ? 1 : i * fac( i - 1); }
```

5-misol. Fibonacci sonlarini hosil qilish dasturini tuzing. Fibonacci sonlari quyidagicha topiladi:

$$f_0 = 1; f_1 = 1; f_2 = f_1 + f_0; \dots\dots$$

$$f_n = f_{n-1} + f_{n-2};$$

Rekursiv jarayonni to'xtatish sharti $n < 2$ deb olinadi. Masalan 9-o'rindagi Fibonacci sonini topish kerak.

```
# include <iostream.h>
void main ( )
{ int n, f ; int fib ( int );
cout << "Nomerni kiriting =";
cin >> n;
f = fib (n);
cout << "Fibonacci soni="<< f << endl;
}
int fib ( int n )
```

```
{ if ( n < 2) return 1; else return ( fib (n-2) + fib (n-1)); }
```

6-misol. $Z = \frac{a^5 + a^{-4}}{2a^n}$ hisoblash dasturi tO'zilsin. Bu yerdagi darajani

hisoblash funksiya sifatida tashkil etilsin. $y = x^n$ ni funksiya deb tashkil etamiz, bu yerda x, n - rasmiy parametrlar

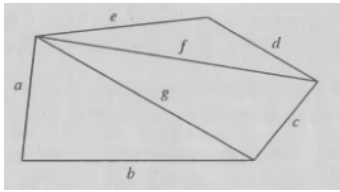
```
# include <iostream.h>
float dar (float x, int n)
{ float y=1;
for (int i=0; i<=n; i++)
    y = y*x;
    return y; }
void main( )
{
int n=3 ; float a, z;
cin>>a;
z = ( dar(a, 5) + dar(1/a, 4))/ (2* dar(a, n)) ;
cout << "z="<<z<<endl; }
```

Nazorat savollari

1. Funktsiyalarni qo'llashdan maqsad?
2. Funktsiyalarga qiymatlarni uzatish va qaytarish?
3. Funksiya – protseduralarni qo'llash?
4. O'zgaruvchilarni adres bo'yicha uzatish?

BAJARISH UCHUN TOPSHIRIQLAR

№	Funksiyaga doir variantlar
1	Sonni o'nlik sanoq sistemasidan o'n oltilik sanoq sistemasiga (hamda teskarisiga) o'tkazuvchi funksiya tuzing.
2	Sonni o'nlik sanoq sistemasidan sakkizlik oltilik sanoq sistemasiga o'tkazuvchi funksiya tuzing.
3	Sonni o'nlik sanoq sistemasidan ikkilik sanoq sistemasiga o'tkazuvchi funksiya tuzing.
4	Sonni o'n oltilik sanoq sistemasidan ikkilik sanoq sistemasiga o'tkazuvchi funksiya tuzing.
5	Sonni o'n oltilik sanoq sistemasidan sakkizlik sanoq sistemasiga o'tkazuvchi funksiya tuzing.
6	Sonni sakkizlik sanoq sistemasidan ikkilik sanoq sistemasiga o'tkazuvchi funksiya tuzing.
7	Q sonini P darajasini topuvchi funksiya tuzing.
8	P sonini oxiridan L sonini qo'shuvchi funksiya tuzing.
9	Berilgan to'g'ri burchakli uchburchakning katetlari yordamida gipotenuzasini topuvchi funksiya tuzing
10	Berilgan ikki nuqtaning koordinatasi asosida ular orasidagi masofani topuvchi funksiya tuzing.
11	Berilgan sondagi qo'shni raqamlarining raqamlarining yig'indisiga teng bo'lgan raqamni o'chirivchi funksiya tuzing.
12	Berilgan N soni ikkita tub sonni yig'indisi bo'lishini tekshiruvchi funksiya tuzing.
13	Berilgan son 11 ga bo'linsa 1 aks holda o'chiqaruvchi funksiya tuzing.
14	Berilgan sonni K o'rindagi raqamini N o'rindagi raqami bilan almashtiruvchi funksiya tuzing.
15	Sonni raqamlarini o'sish (kamayish) tartibida saralovchi funksiya tuzing.

16	Uchburchakni uchta uchining koordinatalari berilgan. Uning yuzasini topuvchi funksiya tuzing.
17	Tomonlari a, b, c, d va e haqiqiy sonlardan iborat bo'lgan beshburchak berilgan (1-rasm). Beshburchakni va undagi uchta uchburchaklarni yuzasini topuvchi alohida funksiyalar tuzing.
	
18	Besh burchakning uchlarining koordinatalari $x_1, y_1, x_2, y_2, \dots, x_5, y_5$ berilgan (1-rasm). Beshburchakdagi uchburchakni uchta uchining koordinatalari berilgan. Uning yuzasini topuvchi funksiya tuzing.
19	Sonni tublikka tekshiruvchi funksiya tuzing va barcha uch xonali tub sonlarni chiqaring.
20	Barcha tub sonlar ichidan “egizak”larini topuvchi funksiya tuzing. Tub sonlar “egizak” deyiladi, agarda ular 3 ga farq qilsa. Masalan, 41 va 43. Uch xonali sonlar ichidan barcha “egizak” larni chiqaring.
21	Ikkita son berilgan. Ularni raqamlari yig'indisi kattasini toping. Sonni raqamlari yig'indisini topuvchi funksiya tuzing.
22	Ikkita son berilgan. Ularning raqamlaridan soni ko'pini toping. Sonni raqamlari sonini topuvchi funksiya tuzing.
23	6 xonali barcha baxtli sonni toping. Agarda 6 xonali sonni dastlabki uchta raqamini yig'indisi oxirgi uchta raqamini yig'indisiga teng bo'lsa baxtli son deyiladi. 6 xonali sonni raqamlarini uchta yig'indisini topuvchi funksiya tuzing.
24	Berilgan sonni palindromlikka tekshiruvchi funksiya tuzing. Masalan, 1221. Son palindrom bo'lsa “Palindrom”, aks holda “Palindrom emas” so'zini chiqaring.
25	Berilgan a va b sonlarini EKUBini topuvchi funksiya tuzing.
26	Berilgan a va b sonlarini EKUKini topuvchi funksiya tuzing.
27	Berilgan a, b va c sonlarini EKUBini topuvchi funksiya tuzing.

28	Soʻz berilgan. Shu soʻzni palindromlikka tekshiruvchi funksiya tuzing. Masalan, kiyik. Soʻz palindrom boʻlsa “Palindrom” aks holda “Palindrom emas” soʻzini chiqaring.
29	Berilgan soʻzning harflari sonini topuvchi funksiya tuzing.
30	Berilgan y (yil), o (oy) va k (kun) sonlari berilgan. Shu sonlarni kiritgan holda qaysi kunga toʻgʻri kelishini aniqlovchi dastur tuzing. Masalan, 2014 2 7 uchun “Juma” chiqishi lozim.

9-Laboratoriya ishi. C ++ tilining grafik imkoniyatlari

Ishning maqsadi

1. C++ tilining grafik imkoniyatlari bilan tanishish.
2. Grafik ma'lumotlarni aks ettirish.
3. Rangli tasvirlarni tashkil etish.
4. Trigonometrik funksiya grafiklarini tashkil etish.

Topshiriq

1. Har bir talaba jurnalidagi tartib raqami bo'yicha vazifalarning dasturini yozadi.
2. Har bir vazifani bajarishda foydalanilgan o'zgaruvchi va funksiyalar haqida hisobotda ko'rsatish lozim.
3. Hisobot shaklida oldin vazifa, uni bajarishda foydalanilgan funksiyalar, dastur matni va bajarishdan hosil bo'lgan natijani keltirish lozim.
4. 9-laboratoriya ishida berilgan barcha vazifalarni bitta hisobot shaklida topshiring.

Hisobot shakli

1. Laboratoriya ishining nomi.
2. Laboratoriya ishidagi topshiriq raqami
3. Topshiriqni bajarishda foydalanilgan funksiyalar tavsifi.
4. Topshiriq kodi.
5. Topshiriq natijasi.
6. Dasturning elektron versiyasi (Dasturlar kompyuterda o'qituvchiga ko'rsatiladi. O'qituvchini talab va takliflariga ko'ra dastur to'g'rilanadi va so'ngra hisobot tayyorlanadi.).

NAZARIY QISM

Borland C++ da grafika

```
#include <graphics.h>
#include <conio.h>
void main()
{
int d = DETECT, m;
initgraph(&d, &m, "c:\\borlandc\\bgi"); // agarda borland C: da bo'lsa
... // grafik amallar
getch();
closegraph();
}
```

DEV C++ da

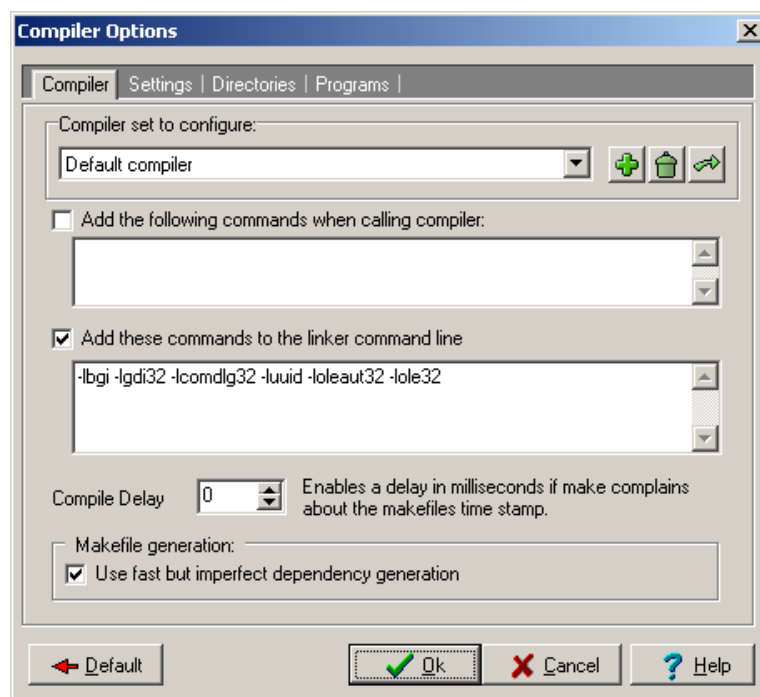
Dev-C++ da grafika bilan ishlash uchun quyidagi so'zlashlarni bajarish lozim: graphics.h va libbgi.a (biblioteka) fayllarni yuklab oling.

1. graphics.h faylni C:\Dev-Cpp\include papkasiga nusxasini ko'chiring.
2. libbgi.a faylni C:\Dev-Cpp\lib papkasiga nusxasini ko'chiring.

Dev-C++ ni ishga tushiring va Tools→Compiler options menyusiga kiring.

Compiler bo'limiga o'ting va Add these commands to the linker command line ga galochka qo'ying va uning oynasiga quyidagi satrni yozing (4.1-rasm):

```
-lbgi -lgdi32 -lcomdlg32 -luuid -loleaut32 -lole32
```

4.1-rasm. Dev C++ da grafika bilan ishlash uchun sozlash.

Chiziq rangini tanlash - `setcolor (rang);`

Rang va bo'yaladigan soha stilini tanlash - `setfillstyle (stili, rangi);`

- 0 – shaffof;
- 1 – to'liq bo'yalgan;
- 3..6 - qiya chiziqli;
- 7..8 – setka;
- 9..11- nuqtali.

Nuqta hosil qilish - `putpixel (x o'qi bo'yicha koordinata, y o'qi bo'yicha koordinata, rang);`

Masalan: `putpixel (150,120, 9);`

To'g'ri chiziq hosil qilish - `line (x1, y1, x2, y2);` x1, y1- chiziq boshi koordinatasi; x2, y2 - chiziq oxiri koordinatasi.

Masalan: `setcolor (10);`

`line (120,100,220,100);`

Siniq chiziqlar hosil qilish

`setcolor (12);` - *chiziq rangi*

`moveto (x1, y1);` - *chiziq boshlanish koordinatasi* `lineto (x2, y2);` - *chiziq singan qismi koordinatasi* `lineto (x3, y3);` - *chiziq oxiri koordinatasi*

to'rtburchak hosil qilish - rectangle (x1, y1, x2, y2); x1, y1- to'rtburchak diagonali boshi koordinatasi; x2, y2 - to'rtburchak diagonali oxiri koordinatasi.

Masalan: setcolor (10); rectangle (100, 100, 200, 200);

to'rtburchak soha hosil qilish

setfillstyle (1, 12); - *stil va rangni tanlash*

bar (x1, y1, x2, y2); - *to'rtburchak soha chizish*; x1, y1- to'rtburchak diagonali boshi koordinatasi; x2, y2 - to'rtburchak diagonali oxiri koordinatasi.

Aylana chizish - circle (x, y, R); x, y – aylana markazi koordinatasi, R – aylana radiusi.

Masalan:

setcolor (LIGHTRED);

circle (120,130,50);

Grafikda matn xosil qilish

setcolor (9);

outtextxy (x, y, "Olim");

x, y – matn hosil bo'ladigan koordinata.

Ranglar:

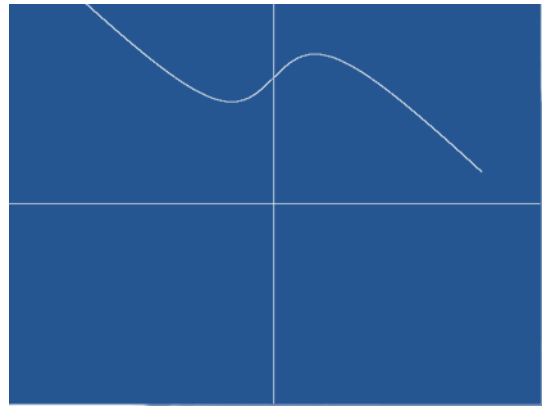
Код	Номи	Код	Номи
0	BLACK	8	DARKGRAY
1	BLUE	9	LIGHTBLUE
2	GREEN	10	LIGHTGREEN
3	CYAN	11	LIGHTCYAN
4	RED	12	LIGHTRED
5	MAGENTA	13	LIGHTMAGENTA
6	BROWN	14	YELLOW
7	LIGHTGRAY	15	WHITE

Namuna. Quyidagi $y=2\arctg(x)-x+3$ funksiya grafigini $x[-5;5]$ oraliqda $n=0.0001$ qadam bilan chizing. Uning C++ tilidagi dasturi quyidagicha.

```

#include<iostream.h>
#include<conio.h>
#include<math.h>
#include <graphics.h>
int main()
{
float y,x;
float z,w;
initwindow(640,480);
//Oyna o'lchami
moveto(320,0); //koordinata gorizontal o'qi boshi
lineto(320,480); //koordinata gorizontal o'qi oxiri
moveto(0,240); //koordinata vertikal o'qi boshi
lineto(640,240); //koordinata vertikal o'qi oxiri
z=320; w=240;
for(x=-5;x<=5;x+=0.0001)
{
y=2*atan(x)-x+3;
moveto(z,w);
putpixel(320+x*50,240-y*50,14);
z=320+x*50;
w=240-y*50;
}
getch();closegraph(); }

```



C++ Builder muhitida grafik shakllarni chizish.

Chizish sirti. Borland C++ Builder 6 muhitida chizish sirti - **TCanvas** sinfi programma ishlash paytida rasm chizish imkonini beradi. Bu sinf obekti sirt bo'yicha ko'chish, grafik primitivlar chizish, rasmlarni va sirtning biror qismini nusxalash, hamda matnni chop qilish imkonini beruvchi xossa va metodlarni o'z ichiga oladi.

Har bir Canvas xossasiga ega komponenta o'z navbatida qalam, kist va shrift obektlarini tarkibiga oladi va mos ravishda Pen, Brush va Font xossalariga ega.

Pen xossasi rangga (*Canvas->Pen-> Color*), chizishning piksellyerdagi qalinlikka (*Canvas->Pen-> Width*), chizilayotgan chiziq toifasiga (*Canvas->Pen->Style*) ega. Chiziq toifasi quyidagi qiymatlarni qabul qilishi mumkin:

psSolid – uzluksiz chiziq (kelishuv bo'yicha);

psDash – tire belgilaridan hosil bo'lgan chiziq;

psDot – nuqtalardan tashkil topgan chiziq;

psDashDot – nuqta va tire ketma-ketliklaridan iborat chiziq;

psDashDotDot – tire va nuqtalardan ketma-ketliklaridan iborat chiziq;

psClear – ko'rinmas chiziq;

psInsideFrame – chizish sirtini chegaralovchi to'g'ri to'rtburchak ichidagi chiziq.

Brush xossasi geometrik shakllar, masalan, to'g'ri to'rtburchak va ellips ichini to'ldirish naqshini aniqlaydi. U quyidagi xossalarga ega:

Canvas->Brush->Color – kist ranggi;

Canvas->Brush->Style – kist toifasini aniqlaydi va u quyidagi qiymatlarni qabul qilishi mumkin:

bsSolid – berilgan rang bilan shakl yuzasi to'liq bo'yaladi;

bsClear -shakl yuzasini bo'yalmaydi;

bsHorizontal - shakl yuzasi parallel chiziqlar bilan to'ldiriladi;

bsVertical - shakl yuzasi vertikal chiziqlar bilan to'ldiriladi;

bsFDiagonal - shakl yuzasi yuqoriga qaragan chiziqlar bilan to'ldiriladi;

bsFDiagonal - shakl yuzasi yuqoriga pastga chiziqlar bilan to'ldiriladi;

bsCross - shakl yuzasi to'r bilan to'ldiriladi;

bsDiagCross - shakl yuzasi egri chiziqlardan hosil bo'lgan to'r bilan to'ldiriladi.

Canvas obektining muhim xossalaridan biri *Canvas->Pixels[x][y]* xossasi bo'lib, u ko'rsatilgan koordinatadagi piksel rangini aniqlaydi. Bu xossa qiymatini o'qish va unga qiymat yozish mumkin.

Geometrik shakllar chizish uchun quyida keltirilgan funksiyalardan foydalanish mumkin:

Arc(int X1, int Y1, int X2, int Y2, int X3, int Y3, int X4, int X4) – yoy chizish. Bu yerda (X1,Y1) va (X2,Y2) – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi. (X3,Y3) va (X4,Y4) nuqtalar mos holda yoy boshlanishi va oxiri koordinatasi.

Chord(int X1, int Y1, int X2, int Y2, int X3, int Y3, int X4, int X4) – ellips vatarini chizish. Bu yerda (X1,Y1) va (X2,Y2) – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi. (X3,Y3) va (X4,Y4) nuqtalar mos holda vatar boshlanishi va oxiri koordinatasi.

Ellipse(int X1, int Y1, int X2, int Y2) – rang bilan to'ldirilgan ellipsni chizish. Bu yerda (X1,Y1) va (X2,Y2) – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi.

Rectangle(int X1, int Y1, int X2, int Y2) – rang bilan to'ldirilgan to'g'ri to'rtburchakni chizish. Bu yerda (X1,Y1) va (X2,Y2) – mos ravishda yoy chiziladigan to'rtburchak sohaning chap yuqori va o'ng past uchlari koordinatasi.

Matematik funksiyalar grafigini chizish

C++ Builder muhitida grafik shakllarni chizish Canvas komponentasi vositasida amalga oshiriladi. Ayrim visual komponentalar bu komponentaga ega. Masalan, TForm, Image, PaintBox va boshqalar.

Formaning (Form1) grafik shakllar chizish sohasi bu mijoz sohasi hisoblanadi va uning o'lchami Form1->ClientWidth (gorizontaliga) va Form1-> ClientHyeight (vertikaliga) bilan aniqlanadi.

Grafika sohasi adreslanuvchi nuqtalarning to'g'ri burchakli massiv ko'rinishida bo'ladi va ixtiyoriy tasvir yonib yoki o'chib turgan piksellar (tasvirning minimal elementi) kompozitsiyasidan hosil bo'ladi. Bu nuqtalar ikkita butun son: nx - nuqtaning gorizontal nomeri va ny nuqtaning vertikal nomeri bilan adreslanadi:

$$0 \leq nx \leq nx_Max; 0 \leq ny \leq ny_Max,$$

bu yerda nx_Max=Form1->ClientWidth va ny=Form1->ClientHyeight.



Grafika sohasining chap yuqori burchagi (0,0) koordinataga ega bo'ladi. (n_x, n_y) qurilma koordinatalari ham deyiladi va ular faqat butun qiymatlarni qabul qiladi.

Kompyuter grafikasida yana ikkita koordinata tizimi qabul qilingan.

Birinchisi (p_x, p_y) - ekran koordinata tizimi bo'lib, unda p_x - gorizontol bo'yicha ekrandagi masofa, p_y -gorizontol bo'yicha. Bu yerda koordinata o'qlari millimetr va dyumlarda o'lchanadi. Ikkinchi koordinata tizimi - dunyoviy (olam) koodinata tizimidir. U (x,y) dekart tizimi bo'lib, programma tuzuvchisi tomonidan aniqlanadi va tasvirlash qurilmasiga bog'liq bo'lmaydi:

$$X_{min} < x < X_{max}; Y_{min} < y < Y_{max}.$$

Dekart koordinatalar tizimida X va Y o'zgarish diapazonlari $(X_{min}, X_{max}, Y_{min}, Y_{max})$ mavhum matematik ikki o'lchamli fazoning to'g'ri burchakli sohasini aniqlaydi. Bu sohani qurilma koordinatasiga akslantirish quyidagicha amalga oshiriladi:

$$n_x = \text{Round}((x - X_{min}) / (X_{max} - X_{min})) * n_{x_Max};$$

$$n_y = \text{Round}((y - Y_{min}) / (Y_{max} - Y_{min})) * n_{y_Max},$$

bu yerda (x,y) - dekart koordinatasidagi nuqta va uning ekrandagi koordinatasi (n_x, n_y) bo'ladi.

Grafik kursor. Grafik kursor matn kursori bajaruvchi ishni bajaradi, lekin u ekranda ko'rinmaydi. Ma'lumki matn kursori ekrandagi belgi o'rnini (80*25 bo'lganida) ko'rsatadi va bu o'rinda belgi chop qilinganda avtomatik ravishda bir o'rin o'ngga suriladi. Grafik kursor esa chiqariluvchi grafik shaklning boshlang'ich koordinatasini ko'rsatadi va uni keyingi joyga (n_x, n_y) nuqtaga ko'chirish uchun maxsus funksiya ishlatiladi: `Form1->Canvas->MoveTo(n_x, n_y);`

Chiziqchlarni chizish. Sohada chiziqni (kesmani) chizish uchun `Form1->Canvas->LineTo()` funksiyasidan foydalaniladi. Masalan, (x_1, y_1) va (x_2, y_2) nuqtalarni tutashtiruvchi kesma chizish uchun quyidagi amallar bajarilishi kerak:

Form1->Canvas->MoveTo(x1,y1) ;

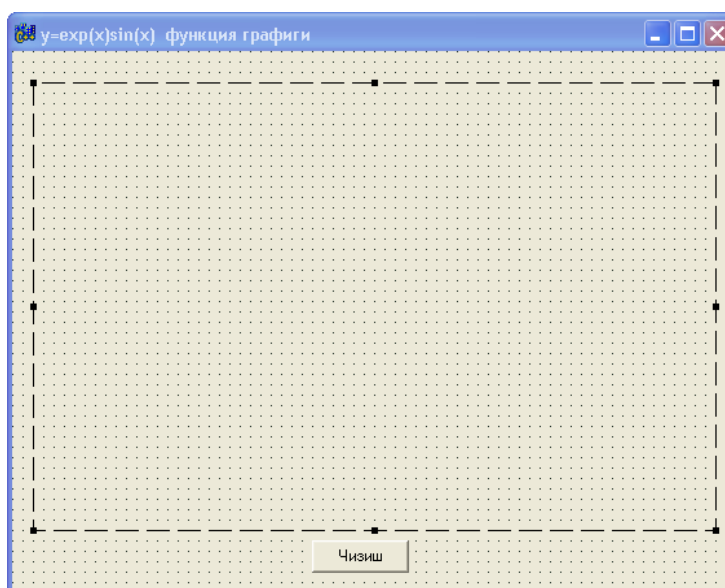
Form1->Canvas->LineTo(x2,y2);

Ektranda ko'p miqdordagi siniq chiziqlardan tashkil topgan shaklni chizish uchun

Canvas->Polyline(Jadval, n);

funksiyasidan foydalaniladi. U berilgan sondagi sonlar juftligi majmuasi bilan aniqlangan siniq chiziqni chizadi. n parametri siniq chiziq tugun nuqtalari soni. Jadval parametri TPoint turida bo'lib, grafik soha nuqta koordinatasini aniqlovchi tuzilmalar massivdir. Siniq chiziq tugun nuqtalari Jadval massivi sifatida beriladi.

Quyida PaintBox (System varagida joylashgan) komponentasi sohasida $y = e^x \sin x$, $x \in [-\pi, \pi]$ funksiya grafigini chizish funksiyasi keltirilgan. Formaga PaintBox komponentasidan tashqari chizishni boshqarish uchun Button1 tugmasi joylashtiriladi:



Funksiya grafigini chizish Button1 tugmasining OnClick hodisasiga bog'lanadi va u quyidagi programma matnida ko'rsatilgan.

```
#include "Unit1.h"
```

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

```
TForm1 *Form1;
```

```
void __fastcall TForm1::Button1Click(TObject *Sender)
```

```
{
```

```

const float Pi=3.1415;
int Xe0,Ye0,Xe,Ye, // Ekran koordinatalari
Chegara_X, Chegara_Y, //Soha chegaralaridagi bo'sh joy o'lchami
Mashtab, // Soha koordinatasining haqiqiysiga munosabati, mashtab
XY;
float h,X,Y; // y=f(x) funksiya va h qadam
Masshtab=20; // Masshtabni tanlash
Chegara_X=10; // Chegaralar
Chegara_Y=10;
h=0.1; // funksiya argumenting h qadam
Xe0 = PictureBox1->Width/2; //Koordinata markazi - Soha markazi tanlandi
Ye0 = PictureBox1->Hyeight/2;
//OX-o'qini chizish
PictureBox1->Canvas->MoveTo(Chegara_X,Ye0);
PictureBox1->Canvas->LineTo(PictureBox1->Width-Chegara_X,Ye0);
// OY-o'qini chizish
PictureBox1->Canvas->MoveTo(Xe0,Chegara_Y);
PictureBox1->Canvas->LineTo(Xe0,PictureBox1->Hyeight-Chegara_Y);
// OX son o'qidagi yo'nalish belgisini chizish (->)
PictureBox1->Canvas->MoveTo(PictureBox1->Width-Chegara_X,Ye0);
PictureBox1->Canvas->LineTo(PictureBox1->Width-Chegara_X-5,Ye0-5);
PictureBox1->Canvas->MoveTo(PictureBox1->Width-Chegara_X,Ye0);
PictureBox1->Canvas->LineTo(PictureBox1->Width-Chegara_X-5,Ye0+5);
// OY son o'qidagi yo'nalish belgisini chizish (^)
PictureBox1->Canvas->MoveTo(Xe0,Chegara_Y);
PictureBox1->Canvas->LineTo(Xe0-5,Chegara_Y+5);
PictureBox1->Canvas->MoveTo(Xe0,Chegara_Y);
PictureBox1->Canvas->LineTo(Xe0+5,Chegara_Y+5);
// OX son o'qini manfiy bo'lagini son kesmalariga bo'lish
XY=0;

```



```

do
{
XY-=1;
Xe=Xe0+(int)(Mashtab*XY);
PictureBox1->Canvas->MoveTo(Xe,Ye0-2);
PictureBox1->Canvas->LineTo(Xe,Ye0+2);
PictureBox1->Canvas->TextOutA(Xe-1,Ye0+2,IntToStr(XY));
}
while(Xe>Chegara_X);
// OX son o'qining musbat bo'lagini son kesmalariga bo'lish
XY=0;
do
{
XY+=1;
Xe=Xe0+(int)(Mashtab*XY);
if(Xe>PictureBox1->Width-Chegara_X)break;
PictureBox1->Canvas->MoveTo(Xe,Ye0-2);
PictureBox1->Canvas->LineTo(Xe,Ye0+2);
PictureBox1->Canvas->TextOutA(Xe-1,Ye0+2,IntToStr(XY));
}
while(1);
// OY son o'qining musbat bo'lagini son kesmalariga bo'lish
XY=0;
do
{
XY+=1;
Ye=Ye0-(int)(Masshtab*XY);
if(Ye<Chegara_Y)break;
PictureBox1->Canvas->MoveTo(Xe0-1,Ye);
PictureBox1->Canvas->LineTo(Xe0+1,Ye);

```

```

PaintBox1->Canvas->TextOutA(Xe0+2, Ye-2, IntToStr(XY));
}
while(1);
// OY son o'qining manfiy bo'lagini son kesmalariga bo'lish
XY=0;
do
{
XY-=1;
Ye=Ye0-(int)(Mashtab*XY);
if(Ye>PaintBox1->Hheight-Chegara_Y)break;
PaintBox1->Canvas->MoveTo(Xe0-1, Ye);
PaintBox1->Canvas->LineTo(Xe0+1, Ye);
PaintBox1->Canvas->TextOutA(Xe0+2, Ye-2, IntToStr(XY));
}
while(1);
X=-Pi-h; // X o'zgaruvchisining chap chegarada qiymatini o'rnatish
X=X+h;
Y= yexp(X)*sin(X); // funksiya qiymati
Xe=Xe0+(int)(Mashtab*X); // X qiymatiga mos ekrandagi piksel
Ye=Ye0-(int)(Mashtab*Y); // Y qiymatiga mos ekrandagi piksel
PaintBox1->Canvas->MoveTo(Xe, Ye); //grafik kursorni o'rnatish
PaintBox1->Canvas->Pen->Color=clRed; // chiziq rangini tanlash
// funksiya grafigini chizish
do
{
X=X+h;
Y= exp(X)*sin(X);
Xe=Xe0 + (int)(Masshtab*X);
Ye=Ye0 - (int)(Masshtab*Y);
if (Xe>Chegara_X && Xe < PaintBox1->Width-Chegara_X

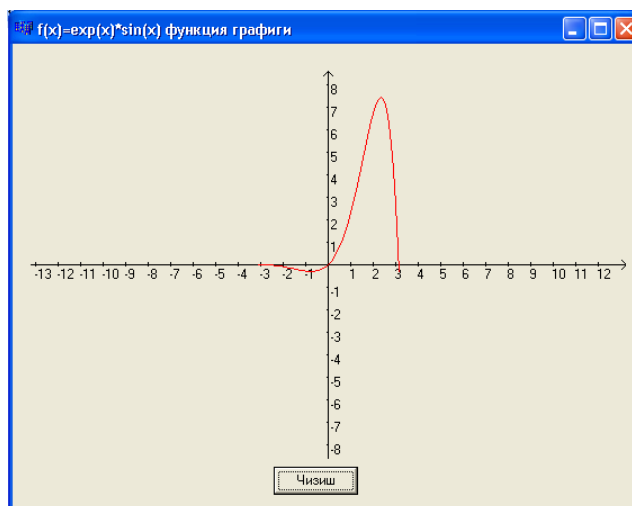
```

```

    && Ye>Chegara_Y && Ye<PaintBox1->Hyeight-Chegara_Y)
    PaintBox1->Canvas->LineTo(Xe,Ye); //Koordinata chegarasida chizish
}
while (X<=Pi);
}

```

Programma ishlashi natijasida Button1 tugmasi (“Chizish”) bosilganda formadagi PaintBox1 komponenta sohasida quyidagi chizma paydo bo’ladi.



9.1 - 9.2-jadvallarda keltirilgan variantlarni Borland C++ Builder 6 da chizing.

Nazorat savollari

1. C/C++ tilida grafik rejimida ishlash uchun qanday direktivalar kerak bo’ladi?
2. Ko’p ishlatiladigan adapterlar haqida ma’lumot bering.
3. Drayver nima va uning vazifasi
4. Oddiy chizmalarni hosil qilish uchun qanday funksiya va proseduralar mavjud?
5. Nuqta, to’g’ri chiziq, aylana, to’g’ri to’rtburchak, ichi bo’yalgan to’rtburchak, ellips, yoy, parallelepiped va shu kabilarni chizish funksiyalari.
6. Sohani bo’yash tuzilmasi.
7. Chizish stillarini o’rnatish.
8. Grafikada matnlarni ishlatish asoslari.
9. Funksiyaning grafigini chizish.

BAJARISH UCHUN TOPSHIRIQLAR

Grafikaga oid variantlar


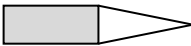
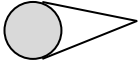
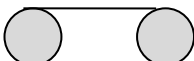
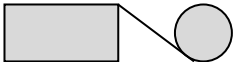
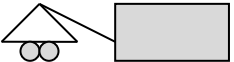
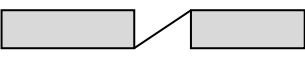
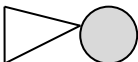

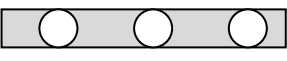

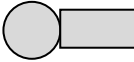
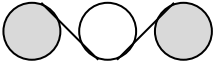

Jadval 9.1

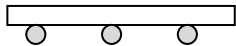

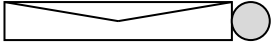
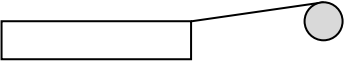
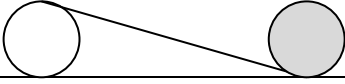
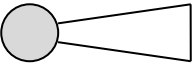
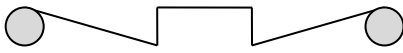
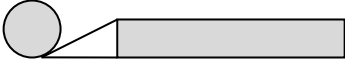
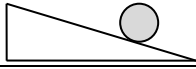
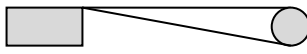

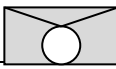
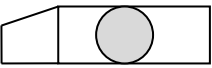

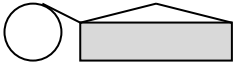

№	Topshiriq sharti
1.	Ot rasmini chizish.
2.	Quyov rasmini chizish
3.	Toshkent kuranti tasvirini yaratish.
4.	Teleminora tasvirini yaratish.
5.	Narda rasmini chizish (toshlari bilan)
6.	Jirafa rasmini chizish.
7.	Fil rasmini chizish.
8.	Oshxona buyumlari to'plami tasvirini yaratish.
9.	“TITANIK” tasvirini yaratish.
10.	“REGISTON” maydoni tasvirini yarating.
11.	O'zbekiston xaritasini viloyatlarni ajratgan holda chizing.
12.	Shaxmat doskasini rasmini chizish (toshlari bilan)
13.	“XUMO” qushi tasvirini chizing.
14.	Toshkent shahri (ixtiyoriy viloyat) xaritasini tumanlarini chegaralari bilan chizing
15.	Miliy kutubxona tasvirini chizing.
16.	“DAMAS” avtomobili tasvirini chizing.
17.	“CAPTIVA” avtomobili tasvirini chizing.
18.	“LASETTI” avtomobili tasvirini chizing.
19.	O'zbekiston havo yo'llari aviakompaniyasi logotipini chizing.
20.	Qanot yozib turgan kapalak tasvirini chizing.
21.	Qaldirg'och rasmini chizing.
22.	“Afrosiyob” poyezdi tasvirini chizing.
23.	Yuk ag'darayotgan yuk mashinasi tasvirini chizing.
24.	Kabutar rasmini chizing.

25.	Burgut rasmini chizing
26.	“Oliy majlis” binosi tasvirini chizing.
27.	Ochilib turgan paxta chanoqlari tasvirini chizing.
28.	O’zbek Davlat sirki binosi tasvirini chizing.
29.	TATU logotipini chizing.
30.	Tez yordam mashinasi tasvirini chizing.

Jadval 9.2

Variantlar.

№	a Shakllar	f(x)	b		
			a	b	n
1		$\text{Sin}x$	$-\pi/2$	$\pi/2$	30
2		$\text{Cos}x$	0	-2π	40
3		$ \text{sin}x + \text{cos}x $	0	π	40
4		$ \text{sin}x - \text{cos}x $	0	π	40
5		$2\text{sin}x+3\text{cos}x$	$-\pi$	π	50
6		$\text{sin}x+\text{cos}2x$	$-\pi$	π	50
7		$2-\text{cos}x$	0	$3\pi/2$	40
8		$\text{Sin}(2x)+\text{cos}x$	0	2π	50
9		$2\text{sin}2x+1$	$-\pi/2$	$\pi/2$	50
10		$\text{sin}x+\text{cos}x-1$	$-\pi$	π	40
11		x^2+2	-3	5	40
12		x^4+1	-1	2	30
13		$10/(1+x^2)$	-3	3	30
14		$1/(x^2-x+1)$	-1	3	40

15		$(x-1)/(x^3+1)$	-1	4	50
16		$1-x^2$	-1	2	50
17		$(x-1)^3$	0	2	20
18		$x x+1 $	-1	2	30
19		$ x+2 ^3$	-3	1	40
20		e^{-x}	-3	1	40
21		e^{x+2}	0	3	40
22		$x e^{-x}$	-3	1	40
23		$x^2 e^{- x }$	-1	3	40
24		$x \sin x$	-1	6	50
25		$x \cos 2x$	-1	4	50
26		$\operatorname{sh} x$	-1	3	50
27		$\operatorname{ch} x - 1$	-1	3	40
28		$\ln x$	1	3	50
29		$\ln(x^2+1)$	1	3	40
30		$e^{0.1} \sin x$	0	$5\pi/2$	50

Foydalanilgan adabiyotlar ro'yxati

1. Nazirov Sh.A., Kabulov R.V., Babajanov M.B., Raxmanov Q.S. C va C++ tili: kasb-hunar kollejlari uchun qo'llanma / O'zbekiston respublikasi Oliv va o'rta maxsus ta'limi markazi. – Toshkent: “Voriz-nashriyot”, 2013.-488b.
2. Култин Н.Б. C/C++ в задачах и премерах. — СПб.: БХВ-Петербург, 2005. -288 с.
3. Romanov V.A. Практикум по программированию на C++: Учебное пособие. СПб.: БХВ-Петербург, Новосибирск: Из-во НГТУ, 2006.- 432с.
4. Вирт Н. Алгоритмы + структуры данных = программа.-М.: Мир, 1985.-405с.
5. Абрамян М.Е. 1000 задач по программированию. ЧАСТЬ II. Минимумы и максимумы одномерные и двумерные массивы, символы и строки, двоичные файлы. Методическое указания для студентов механико-математического физического факультета. Ростов-на-Дону, 2004. – 42с.
6. Златополский Д.М. Сборник задач по программированию. – 2-е изд., перераб. и доп. – СПб.: БХВ-Петербург, 2007.-240с.
7. Raxmanov Q.S., Mo'minov B.B., Qosimova Sh.T., Mahmudov A.Z.. “C/C++ da dasturlash” kursidan laboratoriya ishlari uchun uslubiy ko'rsatma. TATU, Toshkent 2014y. 126 b.
8. Рахманов К.С., Касымова Ш.Т., Гапурова А.А. «Сборник заданий и методических указаний к лабораторным работам по предмету «Программирование на C/C++». ТУИТ. 191 с. Ташкент, 2014.

Internet resurslari

9. acm.tuit.uz.
10. acm.timus.ru
11. neerc.ifmo.ru
12. dastur.uz
13. neerc.ifmo.ru/school
14. acm.dvpion.ru

Mundarija

1-Laboratoriya ishi. Shaxsiy kompyuter arxitekturasi.....	3
2-Laboratoriya ishi. Kompyuterning arifmetik asoslari. Sanoq sistemalari..	12
3-Laboratoriya ishi. Algoritmning turli tuzilmalari.....	36
4-Laboratoriya ishi. Chiziqli algoritm dasturlari	51
5-Laboratoriya ishi. Tarmoqlanuvchi jarayonlarni dasturlash.	66
6-Laboratoriya ishi. Takrorlanuvchi jarayonlarni dasturlash.	78
7-Laboratoriya ishi. Massivlarni tashkil etish.	91
8-Laboratoriya ishi. C++da funksiyalarni tashkil etish.	119
9-Laboratoriya ishi. C ++ tilining grafik imkoniyatlari.....	135
Foydalanilgan adabiyotlar ro'yxati	151

“Informatika” fanidan laboratoriya ishlari uchun uslubiy ko’rsatma

TATU ilmiy-uslubiy kengashining 2015 yil ___ _____dagi
_(___) –sonli majlisida ko’rib chiqildi va nashr etishga
tavsiya etildi.

Mualliflar:

Q.S.Raxmanov,
Sh.T.Qosimova,
Sh.B.Abidova

Ma’sul muharrir:

Q.S.Raxmanov

Korrektor:

Abidova Sh.B.

