

**O`ZBEKISTON RESPUBLIKASI OLIY VA O`RTA
MAXSUS TA'LIM VAZIRLIGI**

**ANDIJON MASHINASOZLIK INSTITUTI
“AXBOROT TEXNOLOGIYALARI” kafedrası**

**“INFORMATIKA VA AXBOROT TEXNOLOGIYALARI”
fanidan**

KYRS ISHI

Mavzu: Ilmiy-texnik masalalarni kompyuterda yechish

Bajardi:

**“Avtomatika va elektrotexnologiya”
fakulteti “TJICHAB” yo’nalishi
1-kurs 2-guruh talabasi**

Usmonov M.

Tekshirdi:

Jamoldinov S.

Andijon- 2013

MAVZU: ILMIY-TEXNIK MASALALARNI KOMPYUTERDA YYECHISH

REJA:

I. Nazariy qism.

1.1. Kirish.

1.2. Masalalarni kompyuterda yechish bosqichlari.

II. Asosiy qism.

2.1. Masalalarni kompyuterda yechishning algoritmlash bosqichi.

2.2. Algoritm tushunchasi va unga misollar

2.3. Algoritmning asosiy xossalari, algoritmni ifodalash usullari va misollar

2.4. Dasturlash tillari va ularni tasniflash, algoritmning asosiy turlari

III. Berilgan masalani algoritmik blok sxemasini va “Pascal” dasturlash tilida tuzilgan dasturini keltiring.

$$P = \prod_{i=1}^5 \prod_{j=1}^6 \frac{i+j}{2a} \quad \text{bu yerda } a = 5.$$

Foydalanilgan adabiyotlar ro'yxati.

KIRISH

Yosh avlodning kompyuter savodxonligi darajasi ilmiy-texnika taraqqiyotini rivojlantirishga katta ta'sir etadi. Shunga ko'ra oliy ta'lim tizimi kurslarida kompyuterlardan foydalanish masalasi dolzarb bo'lib qolmoqda.

Xozirgi kunda ishlab chiqarishning turli texnologiyalaridagi muammolarni hal qilishda zamonaviy hisoblash texnikasi vositalarini keng qo'llash xar bir mutaxassisdan, xox u texnolog bo'lsin xox iqtisodchi bo'lsin albatta zamonaviy kompyuterlardan, hamda axborot texnologiyalaridan kerakli darajada foydalanish mahorati va tegishli ma'lumotga ega bo'lishni talab qiladi.

«Algoritm asoslari va algoritmik tillar» fani orqali ishlab chiqarishning turli sohalariga ta'alluqli bo'lgan masalalarni echish algoritmlarini va dasturini tuzish yo'llari hamda usullarini bilib olinadi.

Informatikada masala yechish tushunchasi deganda axborotlarni qayta ishlab, natijani oldindan belgilangan ma'lum bir ko'rinishga olib kelish tushuniladi.

Kompyuterdan foydalanib masalani yechish - yaratilgan algoritmgaga asoslangan xolda dastlabki ma'lumotlar ustida avtomatik tarzda amallar bajarilib izlangan natija (natijalar) ko'rinishiga keltirish demakdir.

1.2 Masalalarni kompyuterda yechish bosqichlari.

Kompyuterdan foydalanib "ilmiy - texnik masalani yechish" tushunchasi keng ma'nodagi so'z bo'lib, quyidagi bosqichlarga bo'linadi.

Maqsad - bosqichlarning qaysi birlarini mutaxassis kompyuterdan foydalanmasdan va qaysi bosqichlarini kompyuterdan foydalanib bajarishini aniqlash xamda bosqichlarni to'la o'rganib chiqishdan iborat.

Ilmiy - texnik masalalarni (ITM) kompyuterdan foydalanib yechish bosqichlari:

1. Masalaning qo'yilishi va maqsadning aniqlanishi;
2. Masalani matematik ifodalash;
3. Masalani yechish uslubini ishlab chiqish, sonli usullarni tanlash;
4. Masalani yechish algoritmini ishlab chiqish;
5. Ma'lumotlarni tayyorlash va tarkibini aniqlash (tanlash);
6. Dasturlash;
7. Dastur matnini va ma'lumotlarni axborot tashuvchiga o'tkazish;
8. Dastur xatolarini tuzatish;
9. Dasturning avtomatik tarzda kompyuterda bajarilishi;
10. Olingan natijalarni izohlash, tahlil qilish va dasturdan foydalanish uchun ko'rsatma yozish;

"Informatika" kursida 1- 4 bosqichlar qisqa ma'noda, xususiy holatlar, ko'p uchraydigan murakkab bo'lmagan muammolar uchun tushuntiriladi.

Bu bosqichlar tom ma'noda to'laligicha mutaxassislikni egallash davomida maxsus kurslar vositasida o'rgatiladi.

8- va 9-bosqichlarni bajarishda mutaxassis kompyuterdan foydalanadi.

7-bosqichda kompyuterdan foydalanish ham, foydalanmaslik ham mumkin.

ITM ni kompyuterda yechish bosqichlarini aloxida ko'rib chiqamiz.

1-bosqich. MASALANING QO'YILISHI VA MAQSADNI ANIQLASH.

Xalq xo'jaligining muayyan sohasi (texnika, iqtisod, lingvistika, ta'lim va x.k.) bo'yicha ishlayotgan (ishlagan) malakali va etakchi mutaxassis tomonidan

bajariladigan ish, masalani qo'yish va maqsadni aniqlash uchun malakali mutaxassis bir necha kun, oy, xattoki yillab izlanishi mumkin.

Qo'yilgan maqsadni amalga oshirish uchun kerakli ma'lumotlar tarkibi (strukturasi), tuzilishi, ifodalanishi aniqlangan bo'lib, ular orasidagi bog'lanishlar aniq ifodalangan bo'lsa, "masala qo'yilgan" deb aytiladi.

2-bosqich. MASALANI MATEMATIK IFODALASH.

Bu bosqichda masalani yechish uchun kerakli va etarli bo'lgan dastlabki ma'lumotlarning tarkibi, tavsifi, turi, tuzilishi xsobga olingan xolda matematik terminlarda ifodalanadi xamda masalani yechishning matematik modeli yaratiladi.

Buning uchun xar xil (sohasiga qarab) matematik apparat ishlatilishi mumkin.

Masalan iqtisod soxasidagi mutaxassislar - chiziqli dasturlash, dinamik dasturlash, stoxastik dasturlash, bashorat (prognoz) qilish bilan bog'liq masalalarni yechish matematik apparatini bilishlari kerak; texnika soxasidagi mutaxassislar oddiy differentsial tenglamalar va ularning tizimlari, mexanikaning chetki (kraevie) masalalarini, gaz dinamikasiga oid masalalarni, integral ko'rinishdagi masalalarni ifodalash va yechish uchun ishlatiladigan matematik apparatni to'liq tushunib etgan bo'lishi kerak.

Mutaxassis o'z soxasini xar tomonlama yaxshi o'rgangan va amaliy jixatdan puxta o'zlashtirgan va qo'llaniladigan har xil matematik apparatning barcha imkoniyatlarini to'liq tushunib yetgan va amaliyotga qo'llay oladigan bo'lishi kerak.

Bu bosqichda 2 ta asosiy savolga javob topish kerak:

1. Masalani ifodalash uchun qanday matematik strukturalar maqsadga muvofiq keladi?

2. Echilgan o'xshash masalalar bormi?

Tanlangan matematik struktura (apparat)da masalaning elementida ob'ektlari to'la ifodalanishi zarur.

3-bosqich. MASALANI Yechish USULINI ISHLAB CHIQUISH, SONLI USULNI TANLASH.

Agar dastlabki ma'lumotlar bilan izlanayotgan natijalar (miqdorlar, ma'lumotlar) o'rtasida aniq bog'liqlik (qonuniyat) o'rnatilgan bo'lib va masalani yechish uslubi ishlab chiqilgan bo'lsa yoki o'sha bog'lanishni amalga oshirish uchun tayyor sonli usul (lar) tanlab olinib (masala uchun, masalaning bir qismi uchun) masalaning yechish uslubi yaratilgan bo'lsa, "masalani yechish uslubi ishlab chiqilgan" deyiladi.

Bunda: X - dastlabki ma'lumotlar; Y - natija, maqsad funktsiyasi, izlanayotgan miqdor (lar) bo'lsa, ular orasidagi bog'lanish $Y = f(X)$ kabi olinishi mumkin.

f -dastlabki ma'lumotlar bilan natijani bog'lovchi qonuniyat, qoidalar majmuasi, ya'ni X ma'lumotlar ustida bajariladigan amallar ketma-ketligi yoki tanlab olingan usul.

Masalani yechishning ishlab chiqilgan uslubi yoki tanlab olingan usulning to'g'riligi, samaradorligi keyingi bosqichlarda tekshirib aniqlanadi.

4-bosqich. MASALANI YECHISH ALGORITMINI YARATISH.

4-bosqichda asosan masalani yechish algoritmi yaratiladi. Masalani yechish algoritmi kompyuterning imkoniyatlarini, yechish aniqligini xamda masalani kompyuterda yechish vaqtini va qiymatini xisobga olgan xolda yaratilsa maqsadga muvofiq kelgan bo'lar edi.

Masalaning algoritmini yaratishda oraliq ma'lumotlarni iloji boricha kamaytirish, tashqi qurilmalar bilan bo'ladigan aloqalarni minimumga keltirish kerak.

Dasturning samarador va unumdorligi, masalani yechish algoritmining qanchalik puxta tashkil qilinganligiga bog'liq.

3-4 bosqichlar bir-biri bilan jips, mustahkam bog'langan. Ya'ni yaratilgan uslubni har xil usullar bilan amalga oshirish mumkin, shu sababdan masalani yechish uslubi va algoritmining bir nechta variantlari bo'lishi mumkin va keraklisi tanlab olinadi.

Murakkab masalaning algoritmini yaratishda qadamma-qadam oydinlashtirish uslubidan foydalangan ma'qul, har bir qadamda algoritmning tarkibi sodda va tushunarli bo'lib qolishiga erishmoq kerak.

Masalani algoritmlash jarayonida, algoritmning ba'zi bo'laklarini, lavhalarini, mantiqan alohida qismlarini ifodalashda tipik algoritmlar va amaliyotda tekshirilgan algoritmlardan iloji boricha ko'p foydalangan ma'qul.

Algoritmlashda modullik printsiptan foydalanish algoritmni o'qishda va dasturlashda qulayliklar yaratadi. Oxir oqibatda masalani yechish algoritmi ishchi holatga keltiriladi, ya'ni algoritm grafik ko'rinishda biror algoritmik til vositasida ifodalash darajasiga keltiriladi.

Masalani algoritmlash - masalani kompyuterdan foydalanib yechish algoritmini yaratish jarayonidir.

Algoritmlash - masalani yechish bosqichi bo'lib, masalaga qo'yilgan shart va talablar asosida oxirgi natijani, masalaning echimini olish uchun ishlab chiqilgan algoritmlarni yaratish bilan shug'ullanadigan informatikaning bo'limidir.

5-bosqich. MA'LUMOTLARNI TAYYORLASH VA TARKIBINI ANIQLASH.

Ma'lumotlarni tasvirlash usulini tanlash algoritmning bajarilishi bilan chambarchas bog'langan. Shu sababdan ma'lumotni tasvirlashning shunday turini, usulini tanlash kerakki, masalani yechish jarayoni sodda va tushunarli bo'lsin.

Ma'lumotlar oddiy o'zgaruvchilar ko'rinishida (bu xol juda kam uchraydi), massiv ko'rinishida, alohida ma'lumot fayllari (ketma-ket o'qiladigan yoki bevosita o'qiladigan) ko'rinishida axborot tashuvchida joylashgan bo'lishi mumkin.

6-bosqich. DASTURLASH.

Masalani ishchi xolatga keltirilgan yechish algoritmini tanlangan algoritmik til vositasida ifodalash (tavsiflash, tasvirlash) "dasturlash" deyiladi.

Algoritmning xar bir mayda bo'lagi algoritmik tilning operatorlari yordamida, tilning sintaksis va semantika qoidalari asosida yozib chiqiladi. Algoritm mukammal tuzilgan bo'lsa dasturlashda qiyinchilik tug'ilmaydi.

Dasturlash jarayonida quyidagi takliflar inobatga olinsa xatolarni tuzatish jarayoni engillashadi.

1. Dastur umumiy bo'lishi kerak, ya'ni ma'lumotlarni aniq biror turiga bog'liq bo'lmasligi kerak, massivning chegara parametrlarini tekshirmoq lozim. Massiv elementlarining soni 0 yoki 1 bo'lib qolish, yoki yuqori chegarasidan oshib ketish xolati.

2. O'zgarmas kattalik xamda o'zgaruvchi kattalik ko'rinishida ishlatish. (Biror o'zgarmas kattalikni boshqasi bilan almashtirish zarurati bo'lib qolsa, dastur matnini chaqirib o'zgartirish kerak - bu noqulay xolat EXE, COM fayllarida aslo mumkin emas).

Dasturda kiritiladigan ma'lumotlarni nazorat qilish qismi bo'lishi kerak.

3. Dasturdagi arifmetik amallarni kamaytirish va dasturning ishlashini tezlatish uchun:

- darajaga oshirish amallari ko'paytirish amali bilan almashtirilgani ma'qul;
- bir xil ma'lumot bilan xisoblanayotgan arifmetik (algebraik) ifodalarni bir marta xisoblab qiymatini biror o'zgaruvchida saqlab ishlatish.

- takrorlashlarni tashkil qilishda takrorlanishning chegarasini tekshirish uchun ifodalardan emas balki oddiy o'uzgaruvchilardan foydalanish.

- takroriy xisoblashlar tarkibida uchraydigan va takrorlanish davomida qiymatini o'zgartirmaydigan ifodalarni takrorlanishdan tashqarida xisoblash.

4. Dasturning xar bir bo'lagi, moduli qismiga tushuntirishlar yozilgan bo'lishi kerak. Dasturdagi tushuntirishlar, masalani yechish ketma-ketligini ifodalovchi mantiqiy ketma-ketlikdan iborat bo'lmog'i kerak.

Dasturdagi modullar, qismlar aniq ko'rsatilgan bo'lishi kerak. Takrorlanish boshi va takrorlanish oxiri aloxida qatorda turgani ma'qul.

7-bosqich. DASTUR MATNINI VA MA'LUMOTLARNI AXBOROT TASHUVCHIGA O'TKAZISH.

Kompyuter uchun axborot tashuvchi vositalar bo'lib: perfokarta, perfolenta, magnitli tasma, magnitli disk (egiluvchi magnitli disk, magnitli karta), fleshkalar xizmat qilishi mumkin.

Dastur matni alohida maxsus qurilmalar yordamida yoki kompyuterdan foydalanib axborot tashuvchiga o'tkaziladi.

8-bosqich. DASTURNING XATOSINI TUZATISH.

Bu bosqich masalani kompyuterda yechish bosqichlari ichidagi ko'p vaqt talab qiladigan, mutaxassisdan sabr, qanoat, chidam, aql, zakovat, mantiqiy tez fikrlash, algoritmik tilning barcha imkoniyatlarini, tuzatish (otladka) qilish uslubini, yo'llarini, masalaning mag'zini ikir-chikirlarigacha mukammal bilishni talab qiladigan murakkab izlanuvchan jarayondir.

Bu bosqich "dasturni test bo'yicha tekshirish" deb xam yuritiladi. Dasturning to'g'ri ishlashi va yo'l qo'yilgan xatoliklarni aniqlab tuzatish algoritmini yaratishda yo'l qo'yilgan kamchiliklarni bartaraf qilish xamda tanlangan usulning yaroqli yoki yaroqsiz ekanligini aniqlab beruvchi jarayondir.

Test - maxsus tayyorlangan dastlabki malumotlar bo'lib, ular ustida amallar bajarish bilan masalaning echimi-natija olinadi. Test tayyorlash juda murakkab ish bo'lib, qo'lda hisob-kitob ishlarini bajarishni talab qiladi xamda dasturning xamma qismlarini, bo'laklarini, modullarini tekshirish shart.

Dasturning xatosini tuzatish bo'yicha yo'l - yo'riqlar:

1. Maxsus tayyorlangan ma'lumotlar asosida dasturni qo'lda echib chiqish (imkoni bo'lsa) yoki mantiqan alohida bo'lgan bo'laklarini, modullarini qo'lda xisoblash.

2. Dasturni va uning bo'laklarini, modullarini test yordamida tekshirish.

3. Dasturning kerakli joylariga bosib chiqarish buyrug'ini qo'yish (tuzatishlardan keyin olib tashlanadi).

4. Dasturning xatolarini tuzatishda, muloqot rejimida bajarilganda (STOP) to'xtash buyrug'idan foydalanish.

5. Dasturlash tilini va amal bajaruvchi tizimi (AT)ning maxsus xatolarni tuzatish imkoniyatlaridan foydalanish.

6. Xatolarni tuzatish jarayonida kam xajmdagi ma'lumotlar bilan ishlashni tashkil qilish.

9-bosqich. DASTURNING AVTOMATIK TARZDA KOMPYUTERDA BAJARILISHI.

Kompyuter xatolari tuzatilib tayyorlangan dastlabki ma'lumotlardan foydalangan xolda masalaning echimini (echimlarini) avtomatik tarzda xisoblaydi.

Agar natijalar masalaning echimi uchun yaroqli deb topilsa masalani yechish tugallangan xisoblanadi, aks xolda yuqoridagi bosqichlar qaytadan ko'rib chiqiladi.

10-bosqich. OLINGAN MA'LUMOTLARNI IZOHLASH, TAHLIL QILISH VA DASTURDAN FOYDALANISH UCHUN YO'RIQNOMA YOZISH.

Masalani yechish natijasida olingan sonlar yoki sonlar massivi, matnlar yoki matn ko'rinishidagi massivlar xar taraflama izoxlanib, tushuntiriladi. Dasturdan foydalanish uchun ko'rgazma yozish quyidagilarni o'z ichiga oladi:

- Dastur ishlashi uchun ma'lumotlarni tayyorlash usuli, tuzilishi aniq belgilangan;
- Dasturning ishlashi uchun kompyuterni sozlash yo'llari;
- Dasturni ishga tushirish va ishlash paytida bo'ladigan savol-javoblar;
- Dasturni ishlash jarayonida kelib chiqadigan xar xil xolatlarni bartaraf qilish yo'llari aniq va puxta tushunarli qilib yozilgan bo'lishi kerak.

Masalani yechishning uchta bosqichini quyidagi misollarda ko'rib chiqamiz.

1-MISOL.

1. Masalaning qo'yilishi va maqsadning aniqlanilishi. Koptok 29, 5 m / sek tezlik bilan tepaga tik ravishda tepilgan. U qancha balandlikka ko'tariladi? (Havoning qarshiligi xisobga olinmasin).

2. Masalani matematik ifodalash.

Berilgan: $V_0 = 29,5 \text{ m / sek.}$; $V = V_0$.

Koptokni balandlikka ko'tarilish xarakatini ifodalovchi qonuniyat:

$$h = V_0 \cdot t - g \cdot t^2 / 2 \quad (1)$$

bu erda: t - koptokning ko'tarilish vaqti, sek. ; g - erkin tushish tezlanishi ($9,8 \text{ m / sek}$);

3. Masalani yechish usulini ishlab chiqish.

Koptokning tezligi eng yuqori balandlikka etganda nolga teng bo'ladi:

$V = 0$. Fizika kursidan ma'lumki, tezlik yo'ldan vaqt bo'yicha olingan xosila.

$$V = dh / dt. \quad (2)$$

(1) dan xosila olsak

$$V = V_0 - g \cdot t \quad (3)$$

(3) -ni nolga tenglab t ning qiymatini topamiz:

$$t = V_0 / g \quad (4)$$

(4)-dan t ni topib (1) ga qo'yamiz.

2-MISOL.

1. Masalaning qo'yilishi va maqsadni aniqlash.

X o' Y koordinata tekisligida $Y=0$, $X=a$, $X=b$ to'g'ri chiziqlar va $Y = \sqrt{X}$ egri chiziq bilan chegaralangan shaklning yuzasi aniqlansin.

2. Masalani matematik ifodalash.

Masalaning qo'yilishidan ma'lumki bu shakl egri chiziqli trapetsiyadir.

Uning yuzasini topish aniq integral yordamida quyidagicha xisoblanadi:

$$S = \int_a^b \sqrt{x} dx$$

bu erda: a - integralning quyi chegarasi; b - integralning yuqori chegarasi.

3. Masalani yechish usulini ishlab chiqish (tanlash).

Bu turdagi masalalarni yechishda to'rtburchaklar, trapetsiya yoki Simpson taqribiy usullaridan biri tanlab olinadi va yuza xisoblanadi.

2.1. Masalalarni kompyuterda yechishning algoritmlash bosqichi.

“Algoritmlash” deganda masalani biri ketidan boshqasini bajariladigan xamda oldingisining natijalari keyingilarining bajarilishida ishlatiladigan bosqichlar ketma-ketligiga keltirish tushuniladi. Ayni paytda bu bosqichlardagi amallarni kompyuter bajara olishi ko’zda tutilishi kerak.

Kengroq ma’noda qaraydigan bo’lsak algoritmlash, o’zidan oldingi bosqich - masalani yechish usulini tanlash bosqichi xam, o’zidan keyingi bosqich - kompyuterning xususiyatlarini xisobga olgan xolda boshlang’ich, oraliq va natijaviy axborotlarni tuzilishining ifoda shakllarini tanlashni xam o’z ichiga oladi.

Algoritmlash bosqichining natijasi masalani yechish algoritmi bo’ladi, yani bu bosqichda masalani yechish algoritmi ishlab chiqiladi. Bunda masalani matematik qo’yilishi va tanlangan usul qidirilayotgan natijani olishga olib keladigan xarakatlar ketma-ketligini aniqlash uchun asos bo’lib xizmat qiladi.

2.2. “Algoritm” tushunchasi va unga misollar.

Algoritm deb, masalani yechish uchun bajarilishi lozim bo’lgan amallar ketma-ketligini aniq tavsiflaydigan qoidalar tizimiga aytiladi.

Boshqacha aytganda, algoritm –boshlang’ich va oraliq ma’lumotlarni masalani yechish natijasiga aylantiradigan jarayonni bir qiymatli qilib, aniqlab beradigan qoidalarining biror bir chekli ketma-ketligidir.

Buning moxiyati shundan iboratki, agar algoritm ishlab chiqilgan bo’lsa, uni echilayotgan masala bilan tanish bo’lmagan biron bir ijrochiga, shu jumladan kompyuterga xam bajarish uchun topshirsa bo’ladi va u algoritmning qoidalariga aniq rioya qilib masalani echadi.

Masalan, ko’rib o’tilgan birinchi misolni yechish algoritmini quyidagicha bayon qilsa bo’ladi:

1) kompyuter xotirasiga V_0 va g o’zgaruvchilarning sonli qiymatlari kiritilsin;

2) t ning qiymati $t = V_0 / g$ formula bilan xisoblansin;

- 3) h ning qiymati $h = V_0 t - g t^2 / 2$ (1) formula bilan xisoblansin;
- 4) t va h o'zgaruvchilarning sonli qiymatlari ekranga yoki qog'ozga chiqarilsin;
- 5) xisoblash to'xtatilsin.

Masalaning qo'yilishida koptok 29, 5 m /sek bilan tepilsa, degan shart bor edi. ya'ni, $V_0 = 29,5$ va $g = 9,81$ bo'lsa, t va h qancha bo'ladi? (Talabalarning o'ziga yechish taklif etiladi: $t = 3$ sek, $h = 43,35$ m.) Natija xammada bir xil chiqadi.

Ikkinchi misolning yechish algoritmi quyidagicha bo'ladi:

- 1) kompyuter xotirasiga a va b ning qiymati kiritilsin;
- 2) to'g'ri to'rtburchaklar soni n kiritilsin;
- 3) to'rtburchaklar asosi (eni) xisoblansin: $h = (b-a)/n$
- 4) 1-to'rtburchak balandligi (bo'yi) aniqlansin: x_1 qa
- 5) 1-to'rtburchak yuzi xisoblansin: $S_1 = \text{sqr}(x_1) * h$
- 6) S_1 ning qiymati eslab qolinsin;
- 7) 2-to'rtburchakka o'tilsin; $x_2 = x_1 + h$ (balandligi shunga bog'liq)
- 8) 2-to'rtburchak yuzi xisoblansin: $S_2 = \text{sqr}(x_2) * h$
- 9) S_2 ning qiymati S_1 ning qiymatiga qo'shib qo'yilsin va yig'indi eslab qolinsin;
- 10) n -to'rtburchakka o'tilsin: $x_N = x_{(N-1)} + h = b$
- 11) n -to'rtburchak yuzi xisoblansin: $S_n = \text{sqr}(b) * h$
- 12) S_n ning qiymati $S_1, S_2, \dots, S_{(N-1)}$ lar qiymatiga qo'shilsin;

Algoritmni ishlab chiqish uchun avvalo masalaning yechish yo'lini yaxshi tasavvur qilib olish, keyin esa uni formallashtirish, yani aniq qoidalar ketma-ketligi ko'rinishida yozish kerak.

Algoritmni ishlab chiqishda masalani yechish jarayonini shunday formallashtirish kerakki, bu jarayon etarli darajadagi oddiy qoidalarining chekli ketma-ketligi ko'rinishiga keltirilsin.

Masalan, biz ko'pincha ko'p xonali sonlar ustida asosiy arifmetik amallarni bajarishda vatandoshimiz Al-Xorazmiyning IX asrda yaratgan qoidalarini

ishlatamiz. "Algoritm" atamasi xam ana shu buyuk matematik nomidan kelib chiqqan.

2.3. Algoritmning asosiy xossalari, algoritmnı ifodalash usullari va ularga misollar

Algoritm quyidagi asosiy xossalarga ega: uzluklilik, aniqlik, natijaviylik va ommaviylik.

UZLUKLILIK. Dastlabki berilgan malumotlarni natijaga aylantirish jarayoni uzlukli ravishda amalga oshiriladiki, bunda vaqtning xar bir keyingi keladigan daqiqasidagi miqdor (kattalik)larning qiymati vaqtning shundan oldingi daqiqasida bo'lgan miqdorlar qiymatidan ma'lum bir qoidalar bo'yicha olinadi.

ANIQLIK. Algoritmning xar bir qoidasi aniq va bir qiymatli bo'lishi zarurki, bunda vaqtning biror daqiqasida olingan miqdorlar qiymati vaqtning shundan oldingi daqiqasida olingan miqdorlar qiymati bilan bir qiymatli aniqlangan bo'ladi.

NATIJAVIYLIK. Algoritm masalaning echimiga chekli sondagi qadamlar ichida olib kelishi yoki masalani "echib bo'lmaydi" degan xabar bilan tugashi kerak.

OMMAVIYLIK. Masalaning yechish algoritmi shunday yaratilishi kerakki, uni faqat boshlang'ich malumotlar bilan farqlanadigan masalalarni yechish uchun xam qo'llanilishi kerak.

Bunda boshlang'ich malumotlar "algoritmnı qo'llash soxasi" deb ataladigan birorta soxadan olinadi.

Masalan, yuqoridagi 1 - misolda koptok o'rniga boshqa narsani tik irg'itilsa va uning boshlang'ich tezligi malum bo'lsa, shu algoritm bilan u erishadagan balandlik aniqlanadi.

Algoritmnı ishlab chiqishda uni bir necha xil usul bilan ifodalab bersa bo'ladi. Shulardan uchtasi keng tarqalgan. Bular:

1. Algoritmnı oddiy tilda ifodalash;

2. Algoritmni tuzim ko'rinishida ifodalash;
3. Algoritmni maxsus (algoritmik) tilda yozish.

Algoritmni oddiy tilda ifodalash

Algoritmnlarni ifodalashning eng keng tarqalgan shakli - oddiy tilda so'zlar bilan bayon qilishdir. Bu nafaqat hisoblash algoritmilarida, balki hayotiy, turmushdagi "algoritm"larga ham tegishlidir.

Masalan, biror bir taom yoki qandolat mahsulotini tayyorlashning retsepti ham oddiy tilda tavsiflangan algoritmdir. Shaharlararo telefon - avtomat orqali aloqa o'rnatishning o'ziga xos algoritmidan foydalanasiz. Do'kondan yangi kir yuvish mashinasi yoki magnitofon sotib olinsa, ishni foydalanishning algoritmi bilan tanishishdan boshlaymiz.

Masalani kompyuterda yechishda ham, ko'pincha matematika tilini ham o'z ichiga olgan tabiiy tildan foydalanish mumkin. Algoritmning bunday tildagi yozuvi izlanayotgan natijaga olib keladigan amallar ketma-ketligi ko'rinishida bo'lib, odam tomonidan bir ma'noli idrok etilishi kerak. So'zlar bilan ifodalangan har bir amal "algoritmning qadami" deb ataladi. Qadamlar tartib nomeriga ega bo'ladi.

Algoritm ketma-ket, qadam-ba qadam bajarilishi kerak. Agar algoritm matnida "N sonli qadamga o'tilsin" deb yozilgan bo'lsa, bu algoritmning bajarilishi ko'rsatilgan N-qadamdan davom etishini bildiradi.

Ko'rinib turibdiki, yuqoridagi uchchala misol algoritmi ham oddiy tilda yozilgan ekan.

Algoritmnlarni oddiy tilda ifodalash kompyuterga kiritish uchun yaramaydi. Buning uchun algoritmni kompyuter tilida shunday bayon qilish kerakki, masalan kompyuterda yechish jarayonida bu algoritm ishni avtomatik boshqqarib turadigan bo'lsin.

Kompyuter tushunadigan shaklda yozilgan algoritm masalani yechish dasturidir.

Algoritmni oddiy tilda yozishda to'rt xil amaldan: hisoblash, N- qadamga o'tish, shartni tekshirish, hisoblashning oxiri, shuningdek kiritish va chiqarish

amallaridan foydalanilgan maqul. Bular ichida eng ko'p foydalaniladigani hisoblash amalidir.

Algoritmni tuzim ko'rinishida ifodalash.

Nisbatan murakkab masalalarni yechishda algoritmdan muayyan kompyuter tilidagi dasturga o'tish juda qiyin. Bunday bevosita o'tishda algoritmning alohida qismlari orasidagi bog'lanish yo'qoladi, algoritm tarkibining asosiy va muhim bo'lmagan qismlarini farqlash qiyin bo'lib qoladi.

Bunday sharoitda keyinchalik aniqlash va to'g'rilash ancha vaqt talab qiladigan xatolarga osongina yo'l qo'yish mumkin.

Odatda algoritm bir necha marta ishlab chiqiladi, ba'zan xatolarni to'g'rilash, algoritm tarkibini aniqlashtirish va tekshirish uchun bir necha marta orqaga qaytishga to'g'ri keladi.

Algoritm ishlab chiqishning birinchi bosqichida algoritmni yozishning eng qulay usuli - algoritmni tuzim ko'rinishida ifodalashdir.

Algoritm tuzimi - berilgan algoritmni amalga oshirishdagi amallar ketma-ketligining oddiy tildagi tasvirlash elementlari bilan to'ldirilgan grafik tasviridir. Algoritmning har bir qadami tuzimda biror bir geometrik shakl - blok (blok simvoli) bilan aks ettiriladi.

Bunda bajariladigan amallar turiga ko'ra turlicha bo'lgan bloklarga GOST bo'yicha tasvirlanadigan turli xil geometrik shakllar - to'g'ri to'rtburchak, romb, parallelogramm, ellips, oval va hokazolar mos keladi.

Algoritm tuzimlarini qurish qoidalari GOST 19. 002-80 da (xalqaro standart ISO 2636-73 ga mos keladi) qat'iy belgilab berilgan. GOST 19. 003 -80 (ISO 1028-73ga mos) algoritm va dasturlar tuzimlarida qo'llaniladigan simvollar ro'yxatini, bu simvollarning shakli va o'lchamlarini, shuningdek ular bilan tasvirlanadigan funktsiyalarni (amallarni) belgilaydi.

Tuzim blok(simvol)lari ichida hisoblashlarning tegishli bosqichlari ko'rsatiladi. Shu erda har bir simvol batafsil tushuntiriladi.

Har bir simvol (blok) o'z raqamiga ega bo'ladi. U tepadagi chap burchakka chiziqni uzib yozib qo'yiladi. Tuzimdagi grafik simvollar hisoblash jarayonining rivojlanish yo'nalishini ko'rsatuvchi chiziqlar bilan birlashtiriladi.

Ba'zan chiziqlar oldida ushbu yo'nalish qanday sharoitda tanlanganligi yozib qo'yiladi. Axborot oqimining asosiy yo'nalishi tepadan pastga va chapdan o'ngga ketadi. Bu hollarda chiziqlarni ko'rsatmasa ham bo'ladi, boshqa hollarda albatta chiziqlarni qo'llash majburiydir. Blokka nisbatan oqim chizig'i (potok linii) kiruvchi yoki chiquvchi bo'lishi mumkin. Blok uchun kiruvchi chiziqlar soni chegaralanmagan.

Chiquvchi chiziq esa mantiqiy bloklardan boshqa hollarda faqat bitta bo'ladi. Mantiqiy bloklar ikki va o'ndan ortik oqim chizig'iga ega bo'ladi.

Ulardan har biri mantiqiy shart tekshirishining mumkin bo'lgan natijalarga mos keladi.

O'zaro kesiladigan chiziqlar soni ko'p bo'lganda, chiziqlar soni haddan tashqari ko'p bo'lsa va yo'nalishlari ko'p o'zgaraversa tuzimdagi ko'rgazmalik yo'qoladi. Bunday hollarda axborot oqimi chizig'i uzishga yo'l qo'yiladi, uzilgan chiziq uchlariga "birlashtiruvchi" belgisi qo'yiladi.

Agar uzilish bitta sahifa ichida bo'lsa, O belgisi ishlatilib, ichiga ikki tarafga ham bir xil harf-raqam belgisi qo'yiladi.

Agar tuzim bir necha sahifaga joylansa, bir sahifadan boshqasiga o'tish "sahifalararo bog'lanish" belgisi ishlatiladi.

Bunda axborot uzatilayotgan blokli sahifaga qaysi sahifa va blokka borishi yoziladi, qabul qilinayotgan sahifada esa qaysi sahifa va blokdan kelishi yoziladi.

Algoritm tuzimlarini qurishda quyidagi qoidalarga rioya qilish kerak.

Parallel chiziqlar orasidagi masofa 3 mm dan kam bo'lmasligi, boshqa simvollar orasidagi masofa 5 mmdan kam bo'lmasligi kerak. Bloklarda quyidagi o'lchamlar qabul qilingan: $a=10, 15, 20$; $b=1, 5 \cdot a$.

Agar tuzim kattalashtiriladigan bo'lsa, a ni 5 ga karrali qilib oshiriladi. Bu talablar asosan 10-bosqichda, dasturga yo'riqnoma yozishda rioya qilinadi.

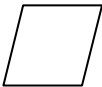
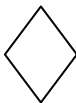
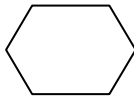
Algoritmnlarni mayda-mayda bo'laklarga ajratishda hech qanday chegaralanishlar qo'yilmagan, bu dastur tuzuvchining o'ziga bog'lik.

Lekin, juda ham umumiy tuzilgan tuzim kam axborot berib, noqulaylik tug'dirsa, juda ham maydalashtirib yuborilgani ko'rgazmalilikka putur etkazadi.

Shuning uchun murakkab va katta algoritmnlarda har xil darajadagi bir nechta tuzim ishlab chiqiladi.

Algoritmning tuzim tarzidagi ifodasining yana bir afzalligi undan uchinchi ko'rinish, ya'ni algoritmik tildagi ifodasi (dastur)ga o'tish ham juda oson bo'ladi. Chunki bunda har bir blok algoritmik tilning ma'lum bir operatori bilan almashtiriladi xolos.

Quyida asosiy bloklar uchun foydalaniladigan shakllar keltirilgan:

ShAKL	Qaysi xolda ishlatiladi	ShAKL	Qaysi xolda ishlatiladi
	Boshlanish va oxirida		Axborotni kiritish va chiqarish
	Xisoblashlar uchun		Natijani chop etish uchun
	Tarmoqlanish shartini tekshirishda		sikl boshlanishida

2.4. Dasturlash tillari va ularni tasniflash, algoritmning asosiy turlari

Algoritmni maxsus tilda ifodalash.

Bu usulda algoritmni ifodalash uchun “dasturlash tillari” deb ataluvchi suniy tillar qo'llaniladi. Buning uchun ishlab chiqilgan algoritm shu tillar yordamida bir manoli va kompyuter tushuna oladigan ko'rinishda tavsiflanishi zarur.

Uning tarkibida cheklangan sondagi sintaksis konstruktsiyalar to'plami bor bo'lib, u bilan algoritm yaratuvchi tanish bo'lishi kerak. Ana shu konstruktsiyalardan foydalanib buyruq va ko'rsatmalar formal ifodalarga o'tkaziladi.

Zamonaviy dasturlash tillari kompyuterning ichki kompyuter tilidan keskin farq qiladi va kompyuter bevosita ana shu tilda ishlay olmaydi. Buning uchun dasturlash tilidan mashina tushunadigan tilga tarjima qiluvchi maxsus dastur - translyatordan foydalaniladi.

Dasturni translyatsiya qilish va bajarish jarayonlari vaqtlarga ajraladi.

Avval barcha dastur translyatsiya qilinib, so'ngra bajarish uslubida ishlaydigan translyatorlar “kompilyatorlar” deb ataladi. Dastlabki tilning har bir operatorini o'zgartirish va bajarishni ketma-ket amalga oshiriladigan translyatorlar “interpretatorlar” deb ataladi.

Dasturlashning ixtiyoriy tili belgilar majmuini va algoritmlarni yozish uchun ushbu belgilarni qo'llash qoidalarini o'z ichiga oladi.

Dasturlash tillari bir biridan alifbosi, sintaksisi va semantikasi bilan ajralib turadi.

Alifbo - tilda qo'llaniladigan ko'plab turli ramziy belgilar (harflar, raqamlar, maxsus belgilar)dir.

Tilning sintaksisi jumlar tuzishda belgilarning bog'lanish qoidalarini belgilaydi, semantikasi esa ushbu jumlarning mazmuniy izohini belgilaydi.

Dasturlash tillari va ularni tasniflash.

Hozirgi kunda dasturlash tillarini u yoki bu belgisi bo'yicha tasniflash mumkin. Dasturlash tilining kompyuterga bog'liqlik darajasi bo'yicha tasniflash eng umumiy hisoblanadi.

Yuqorida aytilgan belgiga qarab, dasturlash tillari kompyutera bog'liq va kompyuterga bog'liq bo'lmagan tillarga bo'linadi.

Kompyuterga bog'lik tillar, o'z navbatida, kompyuter tillari va kompyuterga mo'ljallangan tillarga ajratiladi.

Dasturlash tilining kompyuter tiliga yaqinligi darajasini tariflash uchun til darajasi tushunchasi qo'llaniladi.

Kompyuter tili 0 daraja deb qabul qilingan bo'lib, sanoq boshi hisoblanadi. Odamning tabiiy tili "eng yuqori darajadagi til" deb qaraladi.

Kompyuterga bog'liq bo'lmagan tillar ham ikkita turga bo'linadi: birinchisi protseduraga mo'ljallangan tillar, ikkinchisiga - muammoga mo'ljallangan tillar.

Protseduraga mo'ljallangan tillar turli masalalarni yechish algoritmlarini (protseduralarni) tavsiflashga mo'ljallangan; shuning uchun ular ko'pincha oddiy qilib "algoritmik tillar" deb ataladi.

Ushbu tillar echilayotgan masalalar xususiyatlarini to'la hisobga oladi va kompyuterning turiga deyarli bog'liq emas. Bu xildagi tillar tarkibi kompyuter tiliga qaraganda tabiiy tilga, masalan, ingliz tiliga yaqinroq.

Hozirgi kunda hisoblash, muhandis-texnik, iqtisodiy, matnli va sonli axborotlarni taxlil qilish va boshqa masalalarni yechish tillari malum.

Masalan: FORTRAN tili 1954 yili ishlab chiqilgan bo'lib, FORMula TRANslator - formulalar translyatori degan manoni anglatadi va ilmiy va muhandis - texnik masalalarni hisoblashlarda qo'llaniladi.

ALGOL tili 1960 yili yaratilgan bo'lib, ALGORITMIC Language -algoritmik til degan ma'noni anglatadi va ilmiy-texnik masalalarni hisoblashlarda qo'llaniladi.

KOBOL tili 1959 yili yaratilgan bo'lib, Common Business Oriented Language - "savdo-sotiq masalalariga mo'ljallangan til" degan ma'noni anglatadi. Korxona va tarmoqning moddiy boyligini, moliyasini, ishlab chiqargan

mahsulotini hisobga olish bilan bog'liq iqtisodiy masalalarni yechish uchun qo'llaniladi.

PASKAL tili 1971 yilda e'lon qilingan bo'lib, frantsuz olimi Blez Paskal nomiga qo'yilgan. Turli xildagi masalalar echimini olishda tartiblangan (strukturaviy) dasturlar tuzishda qo'llaniladi.

PL/1 tili 1964 yilda yaratilgan bo'lib, Programming Language/ 1 - 1-tartib raqamli dasturlash tili ma'nosini anglatadi. Ushbu til universal tillar turkumiga kiradi.

Bu tilda ishlab chiqilgan dasturlar kompyuterni yangisi bilan almashtirilganda qaytadan tuzib chiqilishi zarur emas.

BEYSIK (BASIC - Beginner's All Purpose Symbolic Instruction Code - boshlovchilar uchun ko'p maqsadli dasturlash tili) hisoblash algoritmlarini yozish uchun qo'llaniladigan algoritmik tildir. Bu til 1965 yilda Dartmut kolleji xodimlari Kemini va Kurtslar tomonidan ishlab chiqilgan.

Protseduraga mo'ljallangan tillardan masalalarning matematik ifodalari, algoritmlar va dasturlash usullari bilan tanish bo'lgan mutaxassislar foydalaniladilar.

Bunda ulardan kompyuterning tuzilishini mukammal bilish talab qilinmaydi.

Muammoga mo'ljallangan tillar kompyuterda masala yechish usullari va dasturlash usullari bilan tanish bo'lmagan foydalanuvchilar uchun yaratilgandir.

Foydalanuvchi masalani tariflashi, boshlang'ich malumotlarni berishi va natijani chiqarishning talab qilingan ko'rinishini aytishi kifoya.

2.4. Algoritmning asosiy turlari.

Masala echimining algoritmi ishlab chiqilayotgan davrda asosan uch xil turdagi algoritmlardan foydalanib, murakkab ko'rinishdagi algoritmlar yaratiladi.

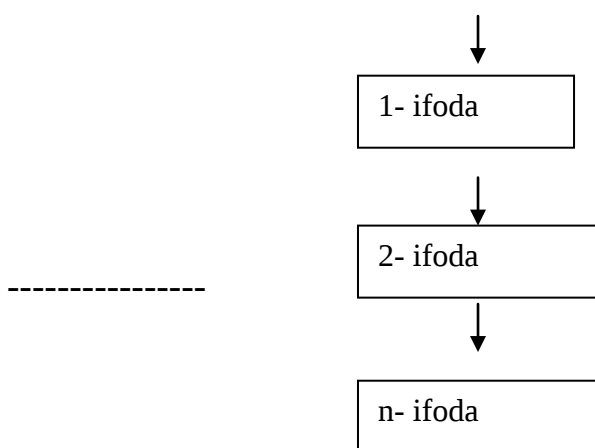
Algoritmning asosiy turlariga chizig'li (a), tarmoqlanadigan (b) va takrorlanadigan (c) ko'rinishlari kiradi.

Murakkab masalalarning echimini olish algoritmlari yuqoridagi turlarining barchasini o'z ichiga olishi mumkin.

Chiziqli turdagi algoritmlarda bloklar biri ketidan boshqasi joylashgan bo'lib, berilgan tartibda bajariladi. Bunday bajarilish tartibi "tabiiy tartib" deb ham yuritiladi.

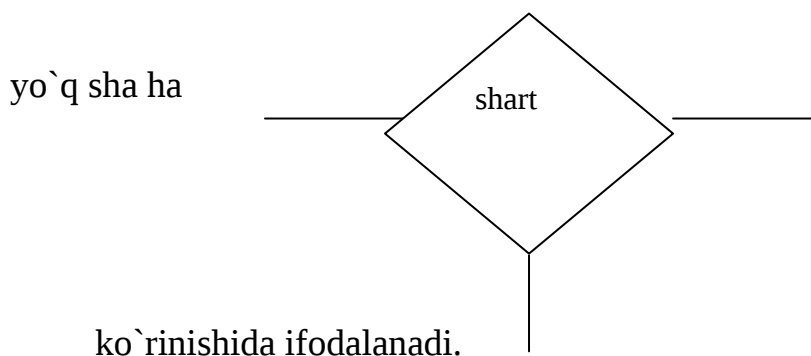
Yuqorida ko'rib o'tilgan birinchi misol chiziqli turdagi algoritmgaga misol bo'ladi. Amalda hamma masalalarni ham chiziqli turdagi algoritmgaga keltirib echib bo'lmaydi.

Chiziqli hisoblash jarayonining tuzimi quyidagicha ko'rinishda ifodalanadi.



Ko'p hollarda biron bir oraliq natijaga bog'liq ravishda hisoblashlar yoki u yoki boshqa ifodaga ko'ra amalga oshirilishi mumkin yani birorta mantiqiy shartni bajarilishiga bog'lik holda hisoblash jarayoni u yoki bu tarmoq bo'yicha amalga oshirilishi mumkin. Bunday tuzilishdagi hisoblash jarayonining algoritmi "tarmoqlanuvchi turdagi algoritmi" deb ataladi.

Algoritmning bu konstruktsiyasi tuzimda



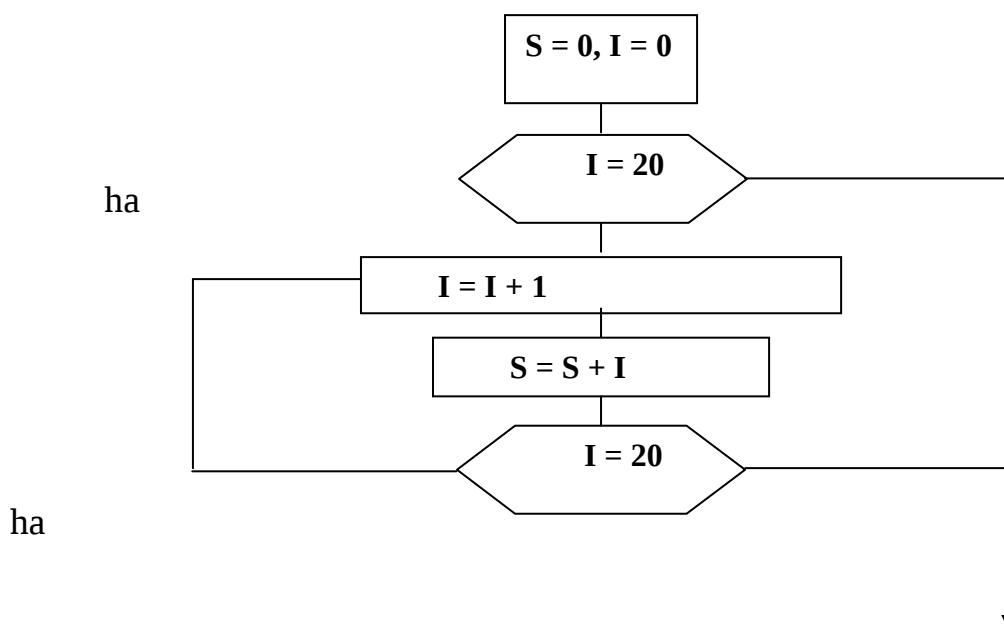
ko'rinishida ifodalanadi.

Ko'pgina hollarda masalalarning echimini olishda bitta matematik bog'lanishga ko'ra unga kiruvchi kattaliklarni turli qiymatlariga mos keladigan qiymatlarini ko'p martalab hisoblash to'g'ri keladi.

Hisoblash jarayonining bunday ko'p martalab takrorlanadigan qismi "takrorlanishlar" deb ataladi. Takrorlanishlarni o'z ichiga olgan algoritmlar "takrorlanuvchi turdagi algoritmlar" deb ataladi. Takrorlanuvchi turdagi algoritmni yozish va chizish o'lchamlarini sezilarli darajada qisqartirish takrorlanadigan qismlarni ixcham ifodalash imkonini beradi.

Yuqoridagi ikkinchi misol takrorlanuvchi turdagi algoritmlarga tegishlidir.

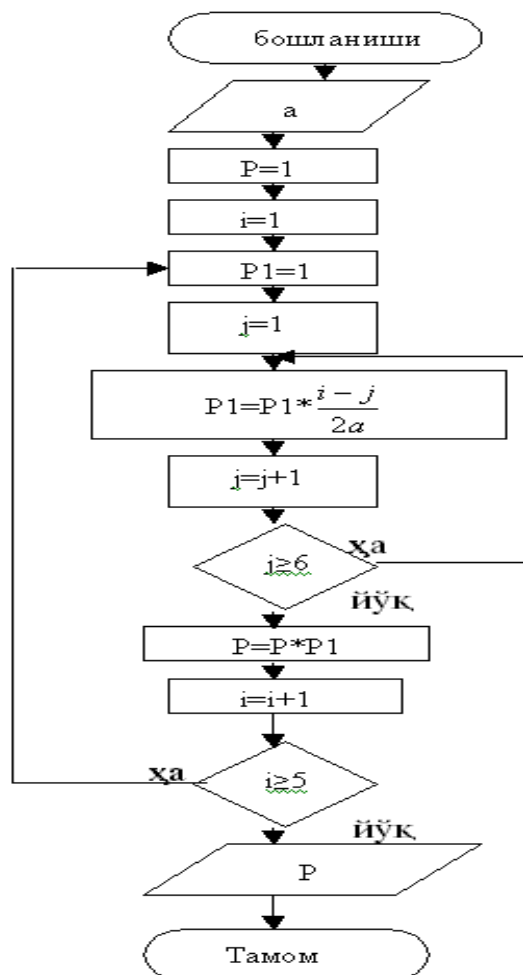
Quyida 1 dan to 20 gacha bo'lgan butun sonlar kvadratlari yig'indisini hisoblash algoritmini tuzim ko'rinishi keltirilgan.



IV. Berilgan masalani algoritmik blok sxemasini va “Pascal” dasturlash

tilida tuzilgan dasturini keltiring. $P = \prod_{i=1}^5 \prod_{j=1}^6 \frac{i+j}{2a}$ bu yerda $a = 5$.

1. Blok-sxemasini tuzamiz:



2. Paskal tilida dasturini tuzamiz:

```
Program kupaytma;
var i, j:integer;
    a,P,P1:real;
begin
  read(a);
  P:=1;
  for i:=1 to 5 do
```

```
begin
    P1:=1;
    for j:=1 to 6 do
        P1:=P1*(i+j)/(2*a);
    P:=P*P1;
    end;
    write('P=',P);
end.
```

Xulosa.

FOYDALANILGAN ADABIYOTLAR

1. O`.T.Haitmatov va b. Informatika va axborot texnologiyalari. O`quv qo`llanma. T. TKTI. 2005 y.
2. O`.T.Haitmatov va b. Informatika va axborot texnologiyalari fanidan laboratoriya ishlarini bajarish ushun uslubiy qo`llanma. T. TKTI. 2005 y.
3. Faronov V.V. Turbo Paskal 7.0. Uchebnoe posobie. M.: Nolidj., 2002g.
4. Aripov M., Xaydarov A. Informatika asoslari T. "O`qituvchi" 2002y.
5. Holmatov T.X.,Toyloqov N.I. Amaliy matematika, dasturlash va kompyuterning dasturiy ta`minoti. T.Mexnat, 2000 y.

AndMI 1-kurs “TJICHAB” yo’nalishi talabasi
M.Usmonovning «Informatika va axborot
texnologiyalari» fanidan “Ilmiy-texnik masalalarni
kompyuterda yechish” mavzusida bajargan kurs
ishiga

T A Q R I Z

Institutining 1-kurs “TJICHAB” yo’nalishi talabasi A.Qobiljonovning
«Informatika va axborot texnologiyalari» fanidan “Ilmiy-texnik masalalarni
kompyuterda yechish” mavzusida bajargan kurs ishida tugallangan ish
hisoblanadi.

Kurs ishida asosan kompyuterda masala yechish bosqichlari,
algoritm lashtirish, dasturlashga oid barcha savollarga javob berilgan.

Kurs ishida avvalo talaba tomonidan mustaqil mavzularni o’zlashtirishga
harakat qilgan.

Shuningdek amaliy topshiriqni ham blok sxemasi va dasturi tuzilgan.
Talabaning bajarigan ishini o’rganib chiqildi va uning himoya vaqtida berilgan
savollarga javobini etiborga olib yahshi baholandi.

Rahbar:

S.Jamoldinov