

ALGORITMLASH VA DASTURLASH USULLARI

Mengatova Xurshida Toshmuxamatovna

TMTI "Avtomatlashtirish va boshqarish" kafedrasini o'qituvchisi:

Tayanch so'z va iboralar: *Algoritm, argument, formula, tenglama, dasturlash, dasturlash tili, algoritmik til, programma, kod, Delphi, Java, C++, Python, windows, JavaScript, Swift, Scala.*

Algoritm va dasturlash usullari

Algoritm so'zi buyuk matematik Al-Xorazmiyning nomi bilan bog'liq bo'lib, u birinchi bo'lib arab raqamlaridan foydalangan holda arifmetik amallarni bajarish qoidasini bayon etdi. Har qanday qo'yilgan masalani kompyuterda yechish uchun oldin uning yechish usulini tanlab, keyin uning algoritmini ishlab chiqish kerak bo'ladi. Demak, hech bir masala yo'qki, uning yechilish yo'llarini bilmasdan va algoritmini tasavvur qilmasdan turib uni kompyuterda yechib bo'lmaydi. Elektron hisoblash mashinalarining vujudga kelishiga qadar algoritimga har xil ta'rif berib kelindi. Lekin ularning barchasi ma'no jihatdan bir-biriga juda yaqin bo'lib, bu ta'rif hozirgi kunda quyidagicha talqin qilinadi.

Ta'rif. Algoritm deb, qo'yilgan masalani yechish uchun ma'lum qoidaga binoan bajariladigan amallarning chekli qadamlar ketma-ketligiga aytiladi.

Har qanday algoritm ma'lum ko'rsatmalarga binoan bajariladi va bu ko'rsatmalarga buyruq deyiladi.

Algoritm quyidagi muhim xossalarga ega:

1. Aniqlik va tushunarlik – deganda algoritmda ijrochiga berilayotgan ko'rsatmalar aniq mazmunda bo'lishi tushuniladi.

2. Ommaviylik – deganda har bir algoritm mazmuniga ko'ra bir turdagi masalalarning barchasi uchun ham o'rinli bo'lishi tushuniladi.

3. Natijaviylik – deganda algoritmda chekli qadamlardan so'ng albatta natija bo'lishi tushuniladi.

4. Diskretlik – deganda algoritmlarni chekli qadamlardan tashkil qilib bo'laklash imkoniyati tushuniladi.

Algoritm turlari

Algoritmning uchta turi bor: chiziqli, tarmoqlanuvchi va takrorlanuvchi.

Chiziqli algoritm – deb hech qanday shartsiz, faqat ketma-ket bajariladigan jarayonlarga aytiladi.

Tarmoqlanuvchi algoritm – deb ma'lum shartlarga muvofiq bajariladigan ko'rsatmalardan tuzilgan algoritimga aytiladi.

Takrorlanuvchi algoritm – deb biron bir shart tekshirilishi yoki biron parametrning har xil qiymatlari asosida algoritmda takrorlanish yuz beradigan jarayonlarga aytiladi.

Algoritmni turli usullarda tasvirlash mumkin. Masalan: so'z bilan ifodalash; formulalarda berish; blok-sxemalarda tasvirlash; dastur shaklida ifodalash va boshqalar.

Algoritmni blok-sxema ko'rinishda tasvirlash qulay va tushunarli bo'lgani uchun ko'p ishlatiladi. Bunda algoritmdagi har bir ko'rsatma o'z shakliga ega. Masalan: parallelogramm ko'rinishdagi belgi ma'lumotlarni kiritish va chiqarish; to'g'ri to'rtburchak belgisi hisoblash jarayonini; romb belgisi shartlarning tekshirilishini bildiradi.

Hayotimizda algoritmni turli sohalarda ba'zan bilgan holda ba'zan esa bilmagan holda ishlatamiz. Algoritm faqat matematik xarakterga ega bo'lmasdan ularni oddiy hayotiy turmushimizda ham ko'p qo'llaymiz. Masalan, ovqat tayyorlash, choy damlash, biror berilgan ishni bajarish va boshqalar. Bu ishlarni bajarishda ma'lum bo'lgan aniq ko'rsatmalarni ketma-ket bajaramiz. Agar bu ko'rsatmalar aniq bir ketma-ketlik tartibida bajarilmasa kerakli natijani ololmaymiz.

Algoritmik tillar

Masalani yechish algoritmi ishlab chiqilgandan so'ng dastur tuziladi. Dastur – bu berilgan algoritmgaga asoslangan biror-bir algoritmik tilda yozilgan ko'rsatmalar, ya'ni buyruqlar yoki operatorlar to'plamidir. Dasturlash – esa bu dastur tuzish jarayoni bo'lib, u quyidagi 4 bosqichlardan iboratdir:

- 1.dasturga bo'lgan talablar;
- 2.qo'yilgan masala algoritmini tanlash yoki ishlab chiqish;
- 3.dastur kodlarini (matnlari, buyruqlarni) yozish;
- 4.dasturni to'g'rilash va test o'tkazish.

Hozirgi kunda juda ko'plab algoritmik tillar mavjud. Ularga dasturlash tillari deb ataymiz. Algoritmik til – algoritmni bir xil va aniq yozish uchun ishlatiladigan belgilashlar va qoidalar tizimidir. Algoritmik til oddiy tilga yaqin bo'lib u matematik belgilarni o'z ichiga oladi. Tuzilgan algoritmni to'g'ridan-to'g'ri mashinaga berib bo'lmaydi, shu sababli yozilgan algoritmnini biror-bir algoritmik tilga o'tkazish zarur. Har qanday algoritmik til o'z qo'llanilish sohasiga ega. Masalan, muxandislik hisob ishlarini bajarishda Paskal, Beysik va Fortran. Iqtisod masalalarini yechishda Paskal va Kobol. Mantiqiy dasturlash uchun Prolog va boshqalar. O'quv jarayonlari uchun Beysik, Paskal va boshqalar.

Paskal, Fortran va Kobol tillari universal tillardan hisoblanadi. Assembler tili mashina tiliga ancha yaqin til bo'lib o'rta darajadagi tildir. Algoritmik til inson tillariga qancha yaqin bo'lsa, u tilga yuqori darajali til deyiladi. Mashina tili esa eng pastki darajali tildir. Algoritm tushunchasi hozirgi zamon matematikasining eng keng qo'llanilayotgan tushunchalaridan biri hisoblanadi. Algoritm so'zi o'rta asrda paydo bo'lgan bo'lib, Al-Xorazmiy nomi bilan bog'liqdir. Hozirgi paytda o'nlik sanoq sistemasida arifmetik amallarni bajarish usullarini hisoblash algoritmi soddagina bo'lgan misol sifatida keltirsa bo'ladi. Ma'lumki, inson kundalik turmushida turli-tuman ishlarni bajaradi. Har bir ishni bajarishda esa bir qancha elementlarni ketma-ket amalga

oshirishga to'g'ri keladi. Mana shu ketma-ketlik yozilsa, u bajariladigan ishning algoritmi bo'ladi.

Algoritm ma'lum bir buyruqlar to'plami bo'lib, bajaruvchi uchun aniq ko'rsatmalarni o'z ichiga mujassamlashtiradi. Ushbu buyruqlar bajaruvchiga ko'rsatilgan maqsadga erishish uchun asos bo'lishi kerak.

Demak, qo'yilgan masalani bajarish ma'lum ketma-ketlikda elementlarni ijro etish orqali erishiladi. Bunda algoritmni bajaruvchi algoritm ijrochisi hisoblanadi. Umuman, uni 2 guruhga ajratish mumkin:

1-guruh algoritmlarining ijrochisi faqat inson bo'lishi mumkin.

2-guruh algoritmlarining ijrochisi ham inson, ham kompyuter bo'lishi mumkin.

Bu guruh algoritmlari ijrochisini kompyuter zimmasiga yuklash mumkin. Buning uchun algoritmni kompyuter tushunadigan biror dasturlash tilida yozib, keyin uning xotirasiga kiritish kifoya. Umuman olganda ijrochi algoritmda mavjud maqsadni bilmaydi. U bevosita keltirilgan buyruqlarni bajaradi. Informatikada algoritmlarning ijrochisi kompyuter deb hisoblanadi. Shunday qilib, biz, algoritm deganda, berilgan masalani yechish uchun ma'lum tartib bilan bajarilishi lozim bo'lgan chekli sondagi ko'rsatmalar ketma-ketligini tushunamiz.

Algoritmlarni yozish uchun qo'llaniladigan tillar algoritmik tillar deb ataladi. Algoritmik tilni kompyuter ham tushunsa, u holda bu til dasturlash tili deb ataladi. Demak, algoritmik yoki dasturlash tillari ham berilgan masalani yechish algoritmining yozish usullaridan biri hisoblanar ekan. Algoritmlarni o'rganish davomida biz quyidagi asosiy tushunchalar bilan tanishamiz: algoritm, blok-sxema, chiziqli algoritmlar, tarmoqli algoritmlar, takrorlash jarayonlari, iteratsion jarayonlar.

Biror masalani kompyuterda yechishda eng muhim va ma'suliyatli ishlardan biri bu masalani yechish algoritmini yaratish bo'lib, bu jarayonda bajarilishi lozim bo'lgan barcha bo'lajak buyruqlar ketma-ketligi aniqlanadi. Algoritmda yo'l qo'yilgan xatoliklar hisoblash jarayonining noto'g'ri bajarilishiga olib keladi, bu esa, o'z navbatida, xato natijaga olib keladi.

Umuman, istalgan masalaning yechimi deyilganda, masalani yechish algoritmi mavjudligi va ushbu algoritm natijaga erishishini ta'minlashi zarurligi tushuniladi.

Masalalarni, ularning qo'yilishi bo'yicha, 4 ta sinfga ajratish mumkin.

1. Aniq ta'riflangan va yechimga erishadigan algoritmlari mavjud masalalar. Ushbu sinfga doir masalalar bevosita to'g'ridan - to'g'ri kompyuterda bajariluvchi dasturlarga aylantirilishi mumkin.

2. Masala qo'yilishida yoki uning yechimida noaniqliklar mavjud bo'lib, algoritmda ushbu noaniqliklarni e'tiborga olish zaruriyati paydo bo'ladi. Bu yerda asosan tashqi muhitning o'zgarishi bilan algoritmning shunga moslanishi nazarda tutiladi.

3. Bilimlarni qayta ishlash doirasida berilgan idrokiy masalalar. Bu yerda mantiqiy tushunchalar bilan ishlay oladigan algoritmik tillarning mavjudligi va ular asosida kompyuter uchun dasturlar yaratish mumkinligi nazarda tutiladi.

4. Inson faoliyatiga bog'liq masalalarni modellashtirishga qaratilgan bo'lib, aniq yechish algoritmlari mavjud bo'lmagan masalalar. Ya'ni inson faoliyatini ma'lum bir model doirasida dasturlash mumkinligi nazarda tutilgan bo'lib, algoritm ushbu faoliyatni to'liq qamrab olmaydi, masalan, shaxmat o'yini.

Berilgan masala algoritmini yozishning turli usullari mavjud bo'lib, ular qatoriga so'z bilan, blok-sxema shaklida, formulalar, operatorlar yordamida, algoritmik yoki dasturlash tillarida yozish kabilarni kiritish mumkin. Endi biror usulda tuzilgan algoritmning ayrim xossalari va ularga qo'yilgan talablarni ko'rib chiqaylik:

1) Algoritm har doim to'liq bir qiymatlidir, ya'ni uni bir xil boshlang'ich qiymatlar bilan ko'p marotaba qo'llash har doim bir xil natija beradi.

2) Algoritm birgina masalani yechish qoidasi bo'lib qolmay, balki turli-tuman boshlang'ich shartlar asosida ma'lum turdagi masalalar to'plamini yechish yo'lidir.

3) Algoritmni qo'llash natijasida chekli qadamdan keyin natijaga erishamiz yoki natijaga erishish mumkin emasligi haqidagi ma'lumotga ega bo'lamiz.

Yuqorida keltirilgan xossalarni umumlashtirib, algoritmning quyidagi asosiy xossalari ta'kidlash zarur.

1. Algoritm boshlang'ich qiymatli argumentlarga ega bo'ladi.

Algoritmni bajarishdan maqsad bu natija olishdir, ya'ni algoritm boshlang'ich qiymatlarni biror usul orqali natijaga aylantiruvchi avtomatdir. Demak, algoritmni qo'yilgan masalaning har xil boshlang'ich qiymatlari uchun qo'llash mumkin. Lekin algoritmning ijrochisi inson bo'lsa, boshlang'ich qiymat ko'rsatilmagan bo'lishi ham mumkin, masalan, "qogozda kvadrat chizing".

Har xil boshlang'ich qiymatlar ishlatilishi mumkin bo'lganligi sababli algoritmning ommaviylik xossasi yuzaga keladi. Lekin ba'zi-bir hollarda faqatgina individual boshlang'ich qiymatga mo'ljallangan algoritm ham mavjud bo'ladi.

2. Algoritm ijrochiga tushunarli bo'lishi lozim, ya'ni ijrochining bazasida mavjud bo'lgan buyruqlar qo'llanilishi zarur. Shu bois, algoritmni yaratish jarayonida ijrochining imkoniyatlari va nozik tomonlari e'tiborga olinishi darkor.

3. Algoritm cheklangan qadamlar ketma-ketligidan iborat bo'ladi va har bir qadam to'liq bajarilgandan so'ng keyingi qadamga o'tiladi. Ushbu xossa algoritmning diskretli xossasi deb ataladi.

4. Algoritm chekli qadamlar bajarilgandan so'ng tugatiladi. Ya'ni chekli qadamlardan so'ng algoritm natijaga erishishi kerak yoki natija olish mumkin emasligi aniqlanishi lozim. Ushbu xossa algoritmning natijaviylik xossasi deb ataladi.

5. Algoritmning har bir qadami aniq ta'riflangan bo'lib, ko'p ma'noli bo'lishi mumkin emas. Ya'ni har bir qadamdan so'ng keyingi qadam aniqlangan bo'lishi yoki algoritm tugatilishi kerak.

Algoritmning ushbu xossasi aniqlik xossasi deb yuritiladi va algoritmni bevosita kompyuterda bajarish uchun imkon beradi.

Ushbu xossaga binoan, bir xil boshlang'ich qiymatlar bilan bajariladigan algoritm doimo bir xil natijaviy qiymatlar bilan tugallanadi.

6. Algoritmning har bir qadami aniq va chekli vaqt davomida bajarilishi lozim. Ushbu xossa algoritmning effektivligi deb yuritiladi.

Algoritmning har bir qadamida bajariladigan ko'rsatma buyruq deb aytiladi. Ushbu buyruqlar nafaqat chekli vaqt davomida, balki qisqa vaqt mobaynida tugatilishi lozim. Dasturlash — kompyuterlar va boshqa mikroprotessorli elektron mashinalar uchun dasturlar tuzish, sinash va o'zgartirish jarayonidan iborat. Odatda dasturlash yuqori saviyali dasturlash tillari (PHP, Java, C++, Python) vositasida amalga oshiriladi. Bu dasturlash tillarining semantikasi odam tiliga yaqinligi tufayli dastur tuzish jarayoni ancha oson kechadi. Dasturlash — bevosita dasturlash va avtomatik dasturlashga bo'linadi. Bevosita Dasturlashda programmaning umumiy sxemasini ishlab chiqishdan kodlash va mashinaga kiritishgacha bo'lgan barcha ishni programmachi bajaradi. Avtomatik dasturlashda esa programmachi faqat programma sxemasini tuzib, uni qisqartirilgan simvolik kurinishda yozadi, programma tuzish va uni kodlash kabi texnikaviy ishlarni esa mashinaning o'zi maxsus dasturlash programmasi yordamida bajaradi.

Dasturlash jarayoni, odatda, quyidagi bosqichlarga bo'linadi:

1. Masalaning qo'yilishi;
2. Masalaning algoritmik tavsifini tuzish;
3. Masalani yuqori darajadagi programma tilida dasturlash;
3. Masalani taxminiy mashina tilida dasturlash;
4. Masalani mashina tilida dasturlash.

Dasturlash tili programmalar tuzishning asosiy vositasidir.

Dasturlashda algoritmlashning asosan 4 turi mavjud:

1. Saralash
2. Qidirish
3. Grafiklar
4. Stringlar

Algoritm — ma'lum bir turga oid masalalarni yechishda ishlatiladigan amallarning muayyan tartibda bajarilishi haqidagi aniq qoida (dastur). Kibernetika va matematikaning asosiy tushunchalaridan biri. O'rta asrlarda sanoqning o'nli tizimi bo'yicha to'rt arifmetik amal bajariladigan qoidani algoritm deb atashgan. "Bu qoidalarni matematikaga IX asrda al-Xorazmiy kiritgan. Yevropada bunday qoidalar uning tugilgan yurtiga nisbatan lotinlashtirilgan (Algorismus yoki Algorithmus shaklida „algorizm“ deyilgan), keyinchalik „algoritm“ga aylangan". Fanda „Yevklid algoritmi“, „G'iyosiddin Koshiy algoritmi“, „Laure algoritmi“, „Markov algoritmi“ deb ataluvchi algoritmlar ma'lum. Algoritm tushunchasi tobora kengayib borib,

kibernetikaning nazariy va mantiqiy asosi hisoblangan algoritmlar nazariyasi paydo bo'ldi. O'zbekistonda bir necha ilmiy tadqiqot muassasalari va hisoblash markazlarida algoritmdan foydalanish sohasida samarali ishlar olib borilmoqda.

Tizimli va amaliy dasturlarni yaratish uchun dasturlash tillaridan foydalaniladi. Dastur yaratish jarayoni katta hajmga ega bo'lib, dasturlash tillarida Tizimli va amaliy dasturlarni yaratish uchun dasturlash tillaridan foydalaniladi. Dastur yaratish jarayoni katta hajmga ega bo'lib, dasturlash tillarida dastur yozishshu jarayonning bir qismi sanaladi. Kompyuter dasturlarini yaratish jarayoni ham kompyuterda masalani yechish jarayoniga o'xshash bo'lib, bir necha bosqichni o'z ichiga oladidastur yozishshu jarayonning bir qismi sanaladi. Kompyuter dasturlarini yaratis jarayoni ham kompyuterda masalani yechish jarayoniga o'xshash bo'lib, bir necha bosqichni o'z ichiga oladi.

U 6 bosqichga bolinadi

Birinchi bosqichda dastur uchun talab aniqlanadi. Ushbu jarayonda dasturning qanday maqsadda qollanilishi, kirish va chiqish ma'lumotlari aniqlanadi. Dastur va resurslarni ishlab chiqish , uni joriy etishning narxi baholanadi.

Ikkinchi bosqichda dastur loyihasi ishlab chiqiladi. Dasturchilar uchun texnik topshiriq va vazifalar shakllantiriladi. Ishchi hujjatlar hamda ish rejasining uchinchi grafigi tuziladi

Uchinchi bosqichda dastur kodi yoziladi. Bu kodlash (dasturlash) jarayoni bo'lib, tuzilgan algoritm dasturlash tilida yoziladi.

To'rtinchi bosqichda kodlash jarayoni tugab, dasturdagi xatoliklarni aniqlash va testlash boshlanadi. Bu jarayonda dasturning to'g'ri tuzilganligi , ishlash samaradorligi va boshqa parametrlariga aniqlik kiritiladi.

Beshinchi bosqichda dastur amaliyotga joriy etiladi. Agar dastur aniq buyurtmachi talabiga muvofiq yaratilgan bo'lsa, u holds bu bosqich eng muhim bosqich sanaladi. Bunda avvalgi dasturda foydalanilgan qurilma va ma'lumotlar yangi dasturga moslanadi. Ushbu dastur bilan ishlovchi mutaxassislar o'qitiladi.

Oltinchi songgi bosqichda qollab- quvvatlash amalga oshiriladi. Bu bosqichda foydalanuvchilarga tavsiyalar beriladi. Shuningdek , foydalanuvchi xohishini inobatga olgan holda, ish jarayonida yuzaga kelgan kamchilik va takliflar asosida dasturga

Dasturlash jarayonida quyi dagi takliflar inobatga olinsa xatolarni tuzatish jarayoni yengillashadi.

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 almash tirish 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'zgaruvchilardan 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 echish 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 alohida qatorda turgani ma'qul.

FOYDALANILGAN ADABIYOTLAR RO'YXATI:

1. A.R.Azamatov "Algoritmlash va dasturlash asoslari"; Mengliyev Sh. "Algoritmlash va dasturlash asoslari";

2. МЕТОД АЛГОРИТМИРОВАНИЯ МАТЕМАТИЧЕСКИХ ЗАДАЧ С ИСПОЛЬЗОВАНИЕМ МАССИВОВ И ЕГО МОБИЛЬНОЕ ПРИЛОЖЕНИЕ И Таиров, Х Менгатова - International Bulletin of Engineering and Technology, 2023

3. BUYUK AJDODLARIMIZNING MEROSI-MA'NAVIYATIMIZ O'CHOG'IDIR

MX Toshmuxamatovna - O'ZBEKISTONDA FANLARARO INNOVATSIYALAR VA ..., 2023

O'ZBEKISTONDA FANLARARO INNOVATSIYALAR VA ILMIY TADQIQOTLAR JURNALI