

8 - Mavzu: Dasturlash tillari va ularning turlari

1. Dasturlash tillari va ularning turlari

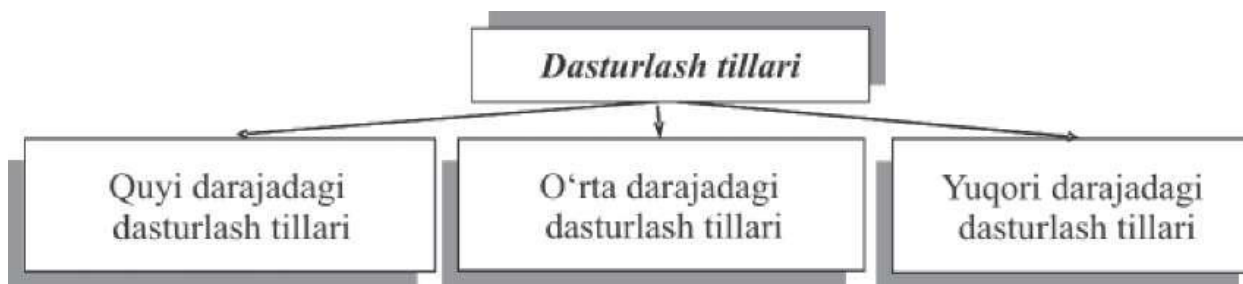
Ma'lumki, kompyuter texnikasidan samarali foydalanish ikki qism texnik va dasturiy ta'minotning uzviyligini talab etadi. Bu uzviylik kompyuter texnik ta'minotining jadal sur'atlar bilan takomillashib borishiga mos dasturiy ta'minotni ham keskin sur'atlar bilan rivojlanishiga sabab bo'ladi, va aksincha. Buning sababi ma'lum, mos dasturiy ta'minotsiz har qanday kompyuter «qimmatbaho o'yinchoq» bo'lib qolaveradi.

Dastur

Ma'lumki, kompyuterda biror masalani hal qilish uchun avval uning qandaydir nusxasi olinadi va algoritmi tuziladi, so'ng mazkur algoritm ma'lum bir qonun-qoidalar asosida kompyuter tushunadigan ko'rsatma va buyruqlar shaklida yoziladi. Hosil bo'lgan matn kompyuter tilida yozilgach, dastur deb ataladi. Demak, dastur — biror masalani yechish uchun kompyuter bajarishi mumkin bo'lgan ko'rsatmalarning izchil tartibi ekan. Kompyuter uchun dastur tuzish jarayoni dasturlash va dastur tuzadigan kishi dasturchi deyiladi.

Dasturlash tili

Kompyuter tushunadigan «til» esa dasturlash tili deb ataladi. Dasturlash tillarini shartli ravishda quyidagi uch guruhga ajratish mumkin:



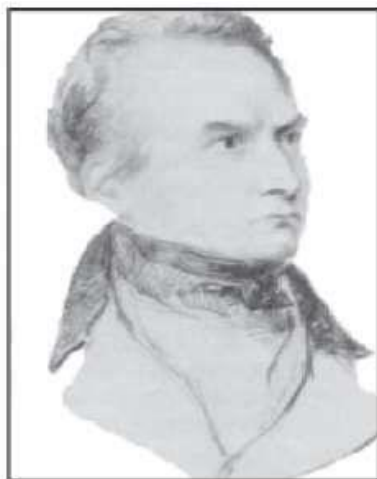
Dasturlash tillari tarixidan. Dasturlash tillari, asosan, ikkinchi jahon urushidan keyin yaratila boshlandi. Ammo uning boshlanish tarixi ancha olis yillarga borib taqaladi.

Arxeologik qazilmalarda topilgan sopol taxtachada bundan 3800 yil oldin (eramizdan avvalgi 1800 yillar) Bobilda foiz bilan bog'liq murakkab amallar algoritmi keltirilgan. Unda aniq masala ishlangan bo'lib, agar bug'doy hosili yiliga 20% dan oshib borsa, uning miqdori ikki marta o'sishi uchun necha yil va oy kerak bo'lish algoritmi tuzilgan.

XIX asr fransuz kashfiyotchisi Jozef Mari Jakkard 1804 yilda yupqa mato ishlab chiqish jarayonida to'quv dastgohlari uchun perfo- kartani eslatuvchi tasma ishlatgan va shu bilan perfokartaga asos solgan edi.

Charlz Bebbij

1836 yilda ingliz olimi Charlz Bebbij hozirgi kompyuterlarning bevosita ajdodi bo'lmish analitik mashina ishlab chiqishga kirishdi va bu masalani nazariy hal qildi. Bu mashinaning asosiy xususiyati uning dastur asosida ishlashi va hisob-kitob natijalarini «eslab» qolishida edi.



Charlz Bebbij

Ogasta Ada Bayron

1843 yilda ingliz matematigi Ogasta Ada Bayron (Lavleys) — shoir lord Bayronning qizi analitik mashina buyruqlar asosida ishlashi lozimligini ta'kidladi. U berilgan shartlar bajarilmagunga qadar qadamlar ketma-ketligini ta'minlovchi buyruqlarni yozdi. Ana shu holat bilan u dasturlash tiliga asos soldi. Mazkur va boshqa kashfiyotlar kompyuter yaratilgach, ularni ishlatish uchun zarur bo'lgan til yaratilishini talab etdi



Ada Bayron

Quyi darajadagi dasturlash tillari

Quyi darajadagi dasturlash tillari kompyuter qurilmalari bilan bevosita bog'liq bo'lib, buyruqlar maxsus raqamlar (kodlar) yordamida yoziladi. Bu kabi buyruqlardan tashkil topgan dasturlar katta hajmli bo'lib, ularni tahrir qilish ancha mushkul ish hisoblanadi. Dastlabki elektron hisoblash mashinalarida («ENIAK», «MESM» va boshqalar) masalalarni yechish uchun ana shunday buyruqlar yordamida dasturlar tuzilgan. Buning asosiy qiyinchiligi — bir tomondan buyruqlarning raqamlar yordamida ifodalanishi bo'lsa, ikkinchi tomondan dasturchidan har bir amalni bajarilishida jamlagichdagi sonli qiymatning qaysi o'zgaruvchiga tegishliligini va boshqa o'zgaruvchilarning qiymatlari qaysi adresda joylashganligini bilish talab etiladi.

O'rta darajadagi dasturlash tillari

Dastur tuzishni osonlashtirish maqsadida inson tiliga yaqin bo'lgan buyruqlar tizimini tuzish va qo'llash masalasi qo'yildi hamda hal etildi. Bu kabi dasturlash tillari o'rta darajadagi dasturlash tillari (ba'zan assemblerlar) deb yuritila boshlandi. Bunday tillarga AVTOKOD-BEMSH, AVTOKOD-MADLEN va boshqalar kiradi. Ular BESM-6, Minsk-22, Minsk-32, IBM- 360 elektron hisoblash mashinalarida ishlatildi.

Yuqori darajali dasturlash tillari

Yuqori darajali dasturlash tillaridagi ko'rsatmalar inson tiliga yaqin bo'lgan so'zlar majmuidan iborat. Ular yordamida amallarni bajarish quyi darajadagilaridan ko'ra yengil bo'lib, biror maxsus ko'rsatma bo'lmasa, dasturchidan adreslar, qurilmalar bilan bevosita bog'liq axborotlarni bilish talab etilmaydi. Bu tilda tuzilgan dasturlarni translyatorlar deb nomlanuvchi maxsus dasturlar kompyuterlar bajara olishi uchun raqamli ko'rinishga o'tkazib beradi.

Keyingi yillarda juda ko'p yuqori darajadagi dasturlash tillari yaratilgan bo'lib, ular qatoriga Paskal, dBase, Ada, KARAT, C++, Delphi, Visual Basic va boshqa tillarni qo'shish mumkin. Hozirgi kunda yaratilayotgan dasturlash tillari biror yo'nalishdagi masalalarni hal qilishga mo'ljallangandir.

Quyidagi jadvalda dasturlash tili rivojlanishi tarixidan qisqacha ma'lumot berilgan. Shuni ham ta'kidlash kerakki, turli rusumdagi kompyuterlar uchun dasturlash tilining ularga moslashtirilgan naqlari ishlab chiqilgan bo'lib, ular bu tilning boshlang'ich naqlidan farq qilishi mumkin.

Plankalkyul

Yuqori darajadagi dastlabki dasturlash tili "Plankalkyul" deb nomlanib, u 1946 - yilda nemis olimi Konrad Suzi tomonidan tuzildi. Bu til o'z vaqtida ma'lum sabablarga ko'ra keng jamoatchilikka tanish emas edi. U 1972 yildan amalda qo'llanila boshlandi

1949 yilda amerikalik Jon Mouchli dasturlashda 8 ta va 10 ta raqamli sanoq sistemalaridan foydalanmaslik taklifi bilan chiqdi. Ana shunga asoslangan

dasturlash tili "Qisqacha kod" nomi bilan Greys Holler tomonidan yaratildi va dastlabki EHM larda ishlatildi.

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

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 echish 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 echish tillari malum.

Dasturlash tillari

FORTTRAN 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. Korxonalar va tarmoqning moddiy boyligini, moliyasini, ishlab chiqargan mahsulotini hisobga olish bilan bog'liq iqtisodiy masalalarni echish uchun qo'llaniladi.

PASKAL

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. Ushbu til universal tillar turkumiga kiradi.

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

BEYSIK

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 echish 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.

Dasturlash tillarining tuzilishi

Avvalgi boblarda turli ijrochilar bilan tanishdik va algoritmlar tuzdik. Endi ba'zi dasturlash tillarida nomlar, ko'rsatmalar, tuzilmalar va boshqalar qanday bo'lishini korib chiqamiz. Biz ko'rmoqchi bo'lgan dasturlash tillarida o'xshashliklar ko'p. Masalan, ularning alifbosi quyidagi asosiy qismlardan iborat:

Lotin alifbosining 26 ta harfi: Aa, Bb, Cc, Dd, Ee, Ff, Gg, Hh, Ii, Jj, Kk, Ll, Mm, Nn, Oo, Pp, Qq, Rr, Ss, Tt, Uu, Vv, Ww, Xx, Yy, Zz;

O'nta arab raqami: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9;

Arifmetik amal belgilari: + (qo'shish), — (ayirish), * (ko'paytirish), / (bo'lish);

Munosabat belgilari: =(teng), <> (teng emas), <(kichik),

<= (katta emas), >(katta), >= (kichik emas);

Maxsus belgilar: (nuqta), (vergul), (nuqtali vergul), (apostrof), (qo'shtirnoq), (undov), (so'roq), % (foiz), \$ (dollar belgisi), @ (tijorat belgisi), & (ampersand), (bo'shliq, ekranda tasvirlanmaydi), (,), {,}, [,] (turli qavslar);

Mantiqiy amallar

AND («VA» — mantiqiy ko'paytirish amali),

OR («YOKI» — mantiqiy qo'shish amali),

NOT («EMAS» — mantiqiy inkor amali).

Yodingizda bo'lsa, nom va qiymati o'zgaradigan miqdorlar haqida aytib o'tgan edik. Yana dasturlash tillarida quyidagilar qo'llaniladi:

Konstantalar (o'zgarmaslar) — dastur ishlashi davomida qiymati o'zgaraydigan miqdorlar;

O'zgaruvchilar — dastur ishlashi davomida qiymati o'zgaradigan miqdorlar;

Algebraik ifodalar — arifmetik amallar bilan bog'langan o'zgarmaslar, o'zgaruvchilar va funksiyalar;

Operatorlar — dasturlash tilining biror tugallangan amalini berish uchun mo'ljallangan buyrug'i, operatorlar BASIC da «: » bilan, PASCAL va DELPHI da «; » bilan ajratiladi;

Funksiya va protseduralar — o'z nomiga ega bo'lgan alohida dastur qismlari (bloklari). Ularga asosiy dasturdan murojaat etiladi;

Nishonlar

Nishonlar — dasturda boshqarish uzatilayotgan operatori ko'rsatadi. Har bir dasturlash tili yuqoridagi tushunchalar bilan bog'liq o'z sintaksisiga, maxsus xizmatchi so'zlariga ega. Dastur yozishdan avval unda ishtirok etadigan miqdorlarni aniqlab olish, o'zgaruvchilarga nom berish va ularni tavsiflash (turini ko'rsatish) kerak bo'ladi. Shundan so'nggina dasturning asosiy qismi boshlanadi.

Har qanday dasturlash tili, odatda, quyidagi ikki qismdan tashkil topadi:

- tavsiflash qismi, PASCAL va DELPHI da VAR xizmatchi so'zi bilan boshlanib va BEGIN xizmatchi so'zidan oldin tugaydi;
- asosiy qism, PASCAL va DELPHI da BEGIN xizmatchi so'zi bilan boshladi hamda END xizmatchi so'zi bilan tugaydi.

O'zgarimas, O'zgaruvchi va Massiv

Dasturlash tillarida asosan uch xil: o'zgarimas, o'zgaruvchi va massiv ko'rinishidagi miqdorlar qo'llaniladi. Ular belgili, satrli, mantiqiy va sonli turdagi qiymatlarni qabul qilishi mumkin.

Belgili o'zgarimaslar ajratish belgisi ichiga olingan bitta belgi - harf, raqam yoki maxsus belgidan iborat

Mantiqiy o'zgarimaslar faqat True (rost) yoki False (yolg'on) qiymatlardan birini qabul qiladi.

Sonli o'zgarimaslar ikki turda — butun yoki haqiqiy bo'lishi mumkin. Haqiqiy sonlar o'z navbatida qo'zg'almas nuqtali va qo'zg'aluvchi nuqtali sonlarga bo'linadi.

Qo'zg'almas nuqtali sonlar — o'nli kasr ko'rinishidagi sonlardir.

O'zgaruvchi miqdorlar

Dasturning bajarilish jarayonida qiymati o'zgaradigan miqdorlar o'zgaruvchi miqdorlar yoki qisqacha o'zgaruvchilar

deyiladi. O'zgaruvchilar barchasida kamida 1 ta belgi, BASIC da 40 ta belgidan oshmaydigan, PASCAL da 63 ta belgidan oshmaydigan, DELPHI da (nom va keyingi yozuvlar sig'ishiga qarab) 1000 ta belgidan ortiq o'z nomiga ega bo'ladi. O'zgaruvchi nomi uchallasida lotin harfi yoki PASCAL va DELPHI da tagchiziq belgisidan boshlanadi. O'zgaruvchi nomida faqatgina raqamlar, lotin harflari va aytib o'tilgan holda, tagchiziq belgisi qo'llanilishi mumkin.

Bu tillarda o'zgaruvchilarning nomida qo'llanilgan katta va kichik lotin harflari farqlanmaydi. Bunday yozish xizmatchi so'zlar uchun ham o'rinalidir.

PASCAL va DELPHI: o'zgaruvchilar dasturning tavsiflash qismida albatta tavsiflanishi, ya'ni ularning turi ko'rsatilgan bo'lishi lozim. Dasturda o'zgaruvchilarni tavsiflash Var xizmatchi so'zi bilan boshlanadi.

BASIC, PASCAL va DELPHI: butun sonli qiymatlar qabul qiladigan o'zgaruvchilar butun sonli o'zgaruvchilar deyiladi. PASCAL va DELPHI da o'xshash 5 tur o'zgaruvchi bo'lib, bir- biridan qabul qiladigan qiymatlarining chegarasi (diapazoni) va kompyuter xotirasidan egallaydigan joyi (hajmi) bilan farqlanadi. Jadvallarda butun sonli o'zgaruvchilarni tavsiflash uchun maxsus so'zlar, ularga mos qiymatlar chegarasi va egallaydigan xotira hajmi keltirilgan:

Jadval ko'rinishidagi miqdorlar

Kundalik hayotimizda ko'p turdagi jadvallardan foydalanamiz: dars jadvali, shaxmat yoki futbol o'yinlari bo'yicha musobaqa jadvali, lotereya jadvali, karra jadvali, matematik jadval va boshqalar. Biz foydalangan tokchalar ham jadvalga misol bo'ladi. Jadvalni tashkil etuvchilar uning elementlari deyiladi.

Jadval ko'rinishidagi miqdorlar bir o'lchovli (chiziqli), ikki o'lchovli uch o'lchovli va hokazo bo'ladi. Biz, odatda, chiziqli va to'g'ri to'rtburchakli jadvallardan foydalanamiz.

Chiziqli jadvallar

Chiziqli jadvallar satr yoki ustun shaklida ifodalanadi. Masalan, sinfingizdagi o'quvchilar ro'yxati sinf jurnalida ustun shaklidagi jadval ko'rinishida yozilgan. O'quvchilarning familiyalari bu jadvalning elementlarini tashkil etadi. Ularning har biri o'z tartib raqamiga ega va har bir tartib raqamga faqat bitta o'quvchining familiyasi mos keladi.

Ikki o'lchovli jadvallar ustunlar va satrlardan tashkil topadi (elektron jadvallarga oid mavzularni eslang). Ularning elementlari ustun va satrlar kesishgan kataklarda joylashadi. Bunday jadvallarda biror elementni ko'rsatish uchun uning nechanchi satr va nechanchi ustunda joylashganligini, ya'ni satr va ustun bo'yicha tartib raqamlarini bilish kerak bo'ladi. Demak, ikki o'lchovli jadvalning har bir elementiga ikkita tartib raqami mos keladi.

Massiv

Dasturlash tillarida jadvallar bilan ishlash uchun massiv tushunchasi kiritilgan. Massiv — jadval ko'rinishidagi miqdor bo'lib, u ma'lum (aniq) sondagi bir turli va tartiblangan (tartib raqamiga ega) elementlar majmuidan iborat. Massiv elementlarining tartib raqami butun sonlarda ifodalanadi. Ular BASIC da manfiy emas, PASCAL va DELPHI da manfiy bo'lishi ham mumkin.

Dasturlash tillarida har bir massiv o'z nomiga ega bo'lib, ularni nomlash o'zgaruvchilarni nomlash kabidir. Masalan: a5, dars_jadvali, tub_sonlar.

Massiv element (tokcha) larning tartib raqami indeks deb ataladi. Indeks BASIC da oddiy qavs, PASCAL va DELPHI da kvadrat qavs ichida ko'rsatiladi. Masalan,

PASCAL va DELPHI da a [5] yozuvi — a nomli massivning beshinchi elementini bildiradi.

Chiziqli dasturlar tuzish

Dasturlash tillari imkoniyatlari va operatorlari haqida ko'p gapirish mumkin. Har bir dasturlash tili uchun alohida qo'llanmalar yozish mumkin.

Chiziqli algoritmlarning dastur shaklida yozilishi chiziqli dastur deb ataladi. Demak, chiziqli dasturdagi amallarni bajarishda hech qanday shart tekshirilmaydi, barcha operatorlar (algoritmdagi ko'rsatmalar) kelish tartibida bajariladi. Yana shuni yodda tutingki, biz DELPHI tilini PASCAL ga o'xshab ketadigan Console Application ilovasida dastur tuzamiz. Shuni inobatga olgan holda dasturning boshida {\$apptype console} direktivasi yozilgan bo'lishi kerak.

O'tish va tarmoqlanish operatorlari

Biz hozirgacha chiziqli, ya'ni buyruqlari ketma-ket bajariladigan dasturlar bilan tanishdik. Lekin ko'pincha berilgan masalani hal qilishda operatorlarning bajarilish tartibini buzishga, ya'ni boshqarishni dastur bo'yicha orqaga yoki oldinga o'tkazish zarur bo'ladi. Buning uchun dasturning boshqarish uzatilayotgan operatoriga nishon qo'yiladi. Nishon o'zgaruvchining nomi kabi lotin harflari va raqamlar yordamida hosil qilinadi

Foydalanilgan adabiyotlar

1. Informatika va informatsion texnologiyalar, M. Aripov va boshqalar. Oliy o'quv yurti talabalari uchun darslik. Toshkent-2019 y.
2. Axborot texnologiyalari, M. Aripov va boshqalar. Oliy o'quv yurti talabalari uchun o'quv qo'llanma. Toshkent-2019 y.
3. Delphi tilida dasturlash asoslari, Sh. Nazirov. Toshkent-2018 y.