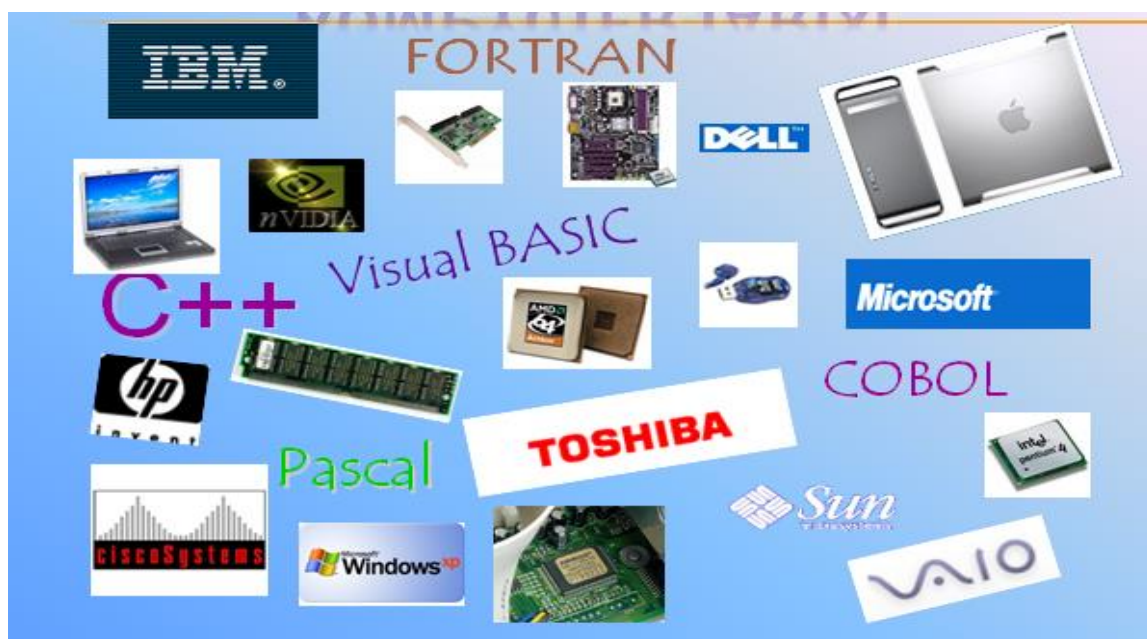


**O'ZBEKISTON RESPUBLIKASI
XALQ TA'LIMI VAZIRLIGI**



**QASHQARYO VILOYAT XALQ TA'LIMI XODIMLARINI
QAYTA TAYYORLASH VA ULARNING MALAKASINI
OSHIRISH INSTITUTI**

TABIY VA ANIQ FANLAR TA'LIMI KAFEDRASI



Paskal dasturlash tili yordamida fizika faniga oid masalalarni

yechish usullari

(metodik qo'llanma)

**TA'LIM VA METODIKA
(AXBOROTNOMA)**

3-songa ilova

Qarshi-2015 yil

K.RASULOVA

**PASKAL DASTURLASH TILI YORDAMIDA FIZIKA FANIGA
OID MASALALARNI YECHISH USULLARI
(metodik qo'llanma)**

Qarshi-2015 yil

TUZUVCHI: **K.RASULOVA, Tabiiy va aniq fanlar ta'limi kafedrası o'qituvchisi**

TAQRIZCHILAR: **N.QAYUMOVA p.f.n. TATU , Qarshi filiali.**
G'JO'RAYEV, Tabiiy va aniq fanlar ta'limi kafedrası katta o'qituvchisi

Tahrir hay'ati:

1. H. Zulfiqorov - Qashqadaryo VXTXQTMOI rektori, v/b
2. X. Xo'jamov – o'quv-metodik ishlar prorektori, v/b
3. Sh. Hamroyev – ma'naviy-ma'rifiy va ilmiy ishlar prorektori, p.f
4. Z. Rahmatova, M. Eshmurodova, X. Nekboyev - ilg'or pedagogik texnologiyalar va tajribalarni joriy etish bo'limi metodistlari

Qo'llanmaning mazmuni va ma'lumotlarning aniqligi uchun muallif shaxsan javobgardir.

Mazkur metodik qo'llanma Tabiiy va aniq fanlar ta'limi kafedrasining 2015 yil 1 maydagi 5-sonli qarori hamda institut ilmiy kengashining 2015 yil 29 maydagi 3 - sonli qarori bilan tasdiqlangan.

MUNDARIJA

Kirish

I-Bob. Dasturlash tillarining sinflari

1.1. Paskal algoritmik tili. Asosiy konstruksiyasi.

Ma'lumotlarning sodda toifa.

1.2. Paskal algoritmik tili dasturining tarkibiy qismi.

II-Bob. Paskal tilida oddiy algoritmlarni dasturlash asoslari.

Dasturlashtirishda bajariladigan harakatlar konsepsiyasi

2.1. Paskal tilida tarmoqlanuvchi jarayonlarni dasturlash.

2.2. Takrorlanish jarayonlarini tashkil qilish. Takrorlanish operatorlari

2.3. Mustaqil ishlash uchun topshiriqlar

Xulosa

Foydalanilgan adabiyotlar

Kirish

Mustaqillikka erishganimizdan so'ng Respublikamizda kompyuterlarni xalq xo'jaligida, umuman jamiyatda qo'llanilishiga katta ahamiyat berila boshlandi.

Hozirda mavjud bo'lgan axborotlarni insoniyat maxsus texnologiya vositasiz qayta ishlash imkoniyatga ega emas. Mutaxassislarning fikricha hozir har 5 yilda mavjud axborot taxminan ikki marotaba ortib bormoqda. Har bir soha mutaxassisi o'z sohasi bo'yicha mavjud axborotlarga ma'lum darajalarda ega bo'lmasa u hozirgi jamiyat talablariga javob bera olmaydi. Demak, u o'z sohasida faoliyat ko'rsatishi uchun mavjud axborotlarni o'zlashtirib borishi, tez suratlar bilan ortib borayotgan axborotlarni o'z vaqtida o'zlashtirib borish texnologiyasiga ega bo'lishi zarur.

Axborotlar oqimini o'zlashtirish, qabul qilish va uzatish jamiyatni rivojlanishida muhim ahamiyatga ega, chunki jamiyatni (davlatni) boshqarishda bo'layotgan xodisa va voqiyalar to'g'risida to'la hamda tez axborot olish davlatni rivojlanishini strategik yunalishlarini aniqlash, zaruriy qarorlar ishlab chiqish imkoniyatini beradi.

Informatsion texnologiya deyilganda axborot oqimlarini qabul qilish, qayta ishlash, uzatish texnologiyasini urganuvchi fan tushuniladi. Bu fan "Informatika" fanini mantiqiy davomi hisoblanib undagi bilimlarga tayanadi.

Xalq xo'jaligidagi turli masalalarni hal etishda algoritmlashtirish nazariyasini yaratish va rivojlantirishga akademik V. Q. Qobulov boshchiligidagi birlashmaning yetakchi olimlari O'zbekistonda kibernetikaning tarkib topishi va rivojlanish bosqichiga ulkan hissa

qo‘shdilar.

Davlat tomonidan tartibga solishning muhimligi va respublikada axborotlashtirish jarayonini tezlashtirish zaruriyatini hisobga olib, O‘zbekiston Respublikasi Vazirlar Mahkamasining 1992 yil 8 dekabr qarori bilan Fan va texnika bo‘yicha Davlat Qo‘mitasi (FTDK) qoshida Axborotlashtirish bo‘yicha bosh boshqarma tuzildi.

Mazkur qarorda belgilab berilgan asosiy vazifa va faoliyat yo‘nalishlari doirasida UzR FTDK tashabbusi bilan axborotlashtirish jarayonini rivojlantirishga yo‘naltirilgan bir qator qonunlar qabul qilindi. Axborotlashtirish haqida (1993 yil, may), EHM va ma’lumotlar bazasi uchun dasturlarni huquqiy muhofazalash haqidagi (1994 yil, may) qonunlar shular jumlasidandir.

UzR FTDK Axborotlashtirish haqida Qonunning qoidalarini bajara borib, 1994 yil dekabrda Vazirlar Mahkamasi O‘zbekiston Respublikasining axborotlashtirish konsepsiyasini ma’qulladi. Ushbu Konsepsiyaning asosiy maqsadi va unda qo‘yilgan masalalar quyidagilardan iboratdir:

- milliy axborot-hisoblash tarmog‘ini yaratish;
- axborotlarga tovar sifatida yondashishning iqtisodiy, xuquqiy va meyoriy xujjatlarini yuritish;
- axborotlarni qayta ishlashning jahon standartlariga rioya qilish;
- informatika industriyasini mujassamlashtirish va rivojlantirish;
- axborotlar texnologiyasi sohasidagi fundamental tadqiqotlarni rag‘batlantirish va qo‘llab-quvvatlash;

➤ informatika vositalari foydalanuvchilarini tayyorlash tizimini muvofiqlashtirish.

Konsepsiyaning asosiy qoidalari hisobga olingan «O‘zbekiston Respublikasining axborotlashtirish dasturi» ishlab chiqildi. U uch maqsadli dasturni o‘z ichiga oladi:

- milliy axborot-hisoblash tarmog‘i;
- kompyuterni matematik va dasturiy ta‘minlash;
- shaxsiy kompyuter.

Mazkur dasturda vazirlik va Mahkama axborot tarmoqlari, Milliy axborot-hisoblash tarmog‘ini yaratish, kompyuter va hisoblash texnikasi vositalarini ishlab chiqarishni tashkil etish, yangi axborot texnologiyalari sohasida kadrlar tayyorlashni takomillashtirish, hujjatlashtirishning meyoriy-uslubiy va huquqiy tizimini yaratish va boshqalar joy olgan.

O‘zbekiston axborot texnologiyalarini tadbiq etish va rivojlantirish uchun talay intellektual imkoniyat va axborot zahiralari ega. Fanlar Akademiyasi, oliy va o‘rta maxsus o‘quv yurtlari, ishlab chiqarish va firmalarda kompyuter texnikasi, aloqa, dasturiy va axborot ta‘minoti, axborot tizimlari bo‘yicha malakali xodimlar ishlamoqda.

Halq xo‘jaligining ushbu yo‘nalishida O‘zbekiston Respublikasi ham yuqorida belgilab berilgan tamoyillarni amalga oshirar ekan, axborotlashgan jamiyat sari shaxdam qadamlar bilan bormoqda.

Buning yorqin dalili sifatida 1997-yil 29-avgustda qabul qilingan «Kadrlar tayyorlash milliy dasturi»ni, O‘zbekiston Respublikasi Oliy Majlisining ikkinchi chaqiriq V sessiyasida Prezident I. Karimov belgilab bergan vazifalar yuzasidan 2001-yil 23-mayda Vazirlar Mahkamasining «2001—2005-yillarda kompyuter va axborot

texnologiyalarini rivojlantirish, «Internet»ning xalqaro axborot tizimlariga keng kirib borishini ta'minlash dasturini ishlab chiqishni tashkil etish chora-tadbirlari to'g'risida»gi qarorini va 2001-yilning may oyida respublikamizda birinchi marta o'tkazilgan Internet festivalini aytib o'tish mumkin.

Yuqoridagi qarorni ijrosini ta'minlash oshirish maqsadida ko'plab ishlar amalga oshirildi, rejalar tuzildi.

Bugungi kunda respublika ta'lim muassasalarida minglab kompyuterlar mavjud bo'lib, ularning asosiy qismini Pentium rusumli zamonaviy kompyuterlar tashkil etadi. Bu kompyuterlar zaruriy dasturiy mahsulotlar bilan ta'minlangan. Ta'lim muassasalarida, shu jumladan, akademik litsey va kasb-hunar kollejlari Internet halqaro axborot tarmog'idan va elektron pochta xizmatidan foydalanuvchilar soni tobora oshib bormoqda. Navbatdagi dolzarb vazifalar sifatida dunyoda mavjud bo'lgan ilg'or va zamonaviy informatsion-pedagogik texnologiyalarni o'rganish, ularni o'quv jarayoniga tadbiq etish, oliy o'quv yurtlari, akademik litseylar va kasb-hunar kollejlari o'rtasida yagona axborot tarmog'ini tashkil etish, o'quv jarayoniga qo'llash uchun dasturiy mahsulotlar ishlab chiqish, virtual kutubxonalar tashkil etish va ularni uzluksiz tarzda jahonda mavjud va respublika ta'lim muassasalarida tayyorlanayotgan elektron darsliklar bilan boyitish kabilar belgilangan.

XXI asrda oliy va o'rta maxsus o'quv yurtlari va xalq ta'limining bitiruvchilari yangi sharoitlarga ijodiy va kasbiy yondashishga tayyorlangan bo'lishlari lozim. Shu sababli Respublikamizda ta'lim sohasida ham boshqa sohalardagi kabi katta o'zgarishlar amalga oshirilmoqda.

O'zbekiston Respublikasi Prezidentining 2002-yil 30 maydagi

«Kompyuterlashtirishni yanada rivojlantirish va axborot-kommunikatsiya texnologiyalarini joriy etish to‘g‘risida»gi farmoni va uni bajarilishini ta‘minlash maqsadida Vazirlar Mahkamasining 2002-yil 6-iyundagi maxsus qarori Respublikamizda informatika va axborot texnologiyalarini rivojlantirishda yana bir muhim bosqich bo‘lib qoldi.

I-Bob. Dasturlash tillarining sinflari.

1.1. Paskal algoritmik tili. Asosiy konstruksiyasi va ma'lumotlarning sodda toifasi.

Algoritmni ifodalash uchun dasturlash tillari deb ataluvchi sun'iy tillar qo'llaniladi. Buning uchun ishlab chiqilgan algoritm shu tillar yordamida bir ma'noli va kompyuter tushuna oladigan ko'rinishda tavsiflanishi zarur. Uning tarkibida cheklangan sondagi sintaksis konstruksiyalar to'plami bor bo'lib, u bilan algoritm yaratuvchi tanish bo'lishi kerak. Ana shu konstruksiyalardan foydalanib, buyruq va ko'rsatmalar formal ifodalarga o'tkaziladi.

Zamonaviy dasturlash tillari kompyuterning ichki mashina tilidan keskin farq qiladi va kompyuter bevosita ana shu tilda ishlay olmaydi. Buning uchun dasturlash tilidan mashina tushunadigan tilga tarjima qiluvchi maxsus dastur - translatordan foydalaniladi. Dasturni translyatsiya qilish va bajarish jarayonlari vaqtlarga ajraladi.

Avval barcha dastur translyatsiya qilinib, so'ngra bajarish uslubida ishlaydigan translatorlar kompilyatorlar deb ataladi.

Dastlabki tilning har bir operatorini o'zgartirish va bajarishni ketma-ket amalga oshiriladigan translatorlar 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

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

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



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

Dasturlash tilining mashina tiliga yaqinligi darajasini tariflash uchun til darajasi tushunchasi qo'llaniladi. Mashina tili 0 daraja deb qabul qilingan bo'lib, sanoq boshi hisoblanadi. Odamning tabiiy tili eng yuqori darajadagi til deb qaraladi.

10

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 yechilayotgan masalalar xususiyatlarini to'la hisobga oladi va kompyuterning toifasiga deyarli bog'liq emas. Bu xildagi tillar tarkibi mashina tiliga qaraganda tabiiy tilga, masalan, ingliz tiliga yaqinroq. Hozirgi kunda hisoblash, muhandis-texnik, iqtisodiy, matnli va sonli axborotlarni tahlil qilish va boshqa masalalarni yechish tillari ma'lum. Masalan: Fortran tili 1954 yili ishlab chiqilgan bo'lib, Formyla Translator -formulalar translatatori degan manoni anglatadi hamda ilmiy va muhandis - texnik masalalarni hisoblashlarda ishlatiladi.

Algol tili 1960 yili yaratilgan bo'lib, Algorithmic Language - algoritmik til degan ma'noni anglatadi va ilmiy-texnik masalalarni hisoblashlarda ishlatiladi.

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 ishlatiladi.

Paskal tili 1971 yilda e'lon qilingan bo'lib, fransuz olimi Blez Paskal nomiga qo'yilgan. Turli xildagi masalalar yechimini olishda tartiblangan (strukturaviy) dasturlar tuzishda ishlatiladi.

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 Sumbolic Instruction Code - boshlovchilar uchun ko'p maqsadli sistemalik instruksiylar kodi) hisoblash algoritmlarini yozish uchun qo'llaniladigan algoritmik til. Bu til 1965 yilda Dartmut kolleji xodimlari Kemini va Kurslar tomonidan ishlab chiqilgan.

Protseduraga mo'ljallangan tillardan masalalarning matematik ifodalari, algoritmlar va dasturlash usullari bilan tanish bo'lgan mutaxassislar foydalanadilar. 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 ta'riflashi, boshlang'ich ma'lumotlarni berishi va natijani chiqarishning talab qilingan ko'rinishini aytishi kifoya.

Paskal algoritmik tili. Asosiy konstruksiyasi.

Ma'lumotlarning sodda toifasi.

XVII asrda yashagan va dunyoda birinchi hisoblash mashinasini yaratgan fransuz olimi Blez Paskal sharafiga shunday deb nomlangan Paskal tili Shveytsariyaning Syurix shahridagi Oliy texnika maktabining professori Niklaus Virt tomonidan 70-yillarda yaratilgan bo'lib, 1979 yilda standart Paskal deb tasdiqlangan.

O'zining soddaligi, mantiqiyligi va samaraliligi tufayli bu til butun dunyoga tezda tarqaldi. Hozirgi paytda barcha hisoblash mashinalari, xususan, mikrokompyuterlar ham shu tilda ishlash imkoniyatiga ega. Dasturlar matnining to'g'riligini osonlik bilan tekshirish mumkinligi, ularning ma'nosi yaqqol ko'zga tashlanishi va oddiyligi bilan ajralib turadi.

Paskal tili ancha murakkab va ko'p vaqt oladigan hisoblash ishlarini bajarishga mo'ljallangan tarkiblashtirilgan dasturlar tuzishga imkon beradi. Yana bir afzalligi shundan iboratki, foydalanuvchi xatolikka yo'l qo'ymasligi uchun yoki xato yozib qo'ygan bo'lsa, tez tuzatib olishi uchun dasturda ishlatilgan o'zgaruvchilar oldindan qaysi turga (toifaga) mansub ekanligi belgilab qo'yilgan bo'ladi. Shu bilan birga dasturning barcha elementlari haqida ma'lumot tavsiflash bo'limida mujassamlashgan bo'ladi. Operatorlar soni esa minimal darajada kamaytirilgandir.

Paskal algoritmik tilining alifbosi quyidagilardan iborat:

- 26 ta lotin harflari;
- 0 dan 9 gacha arab raqamlari;
- 32 ta kirill harflari;
- maxsus belgilar - (Q,-,*,G',:,,[,],q,>,<) ni o'z ichiga oladi.

Paskal tilida so'z deb bir nechta belgilar ketma-ketligi tushuniladi. Xizmatchi so'z deb Paskal tilidagi standart nom tushuniladi. Bu nom maxsus ma'noni anglatadi va uni ma'lumotlarga berib bo'lmaydi. Masalan, Program, Begin, End va h.k.

Paskal tilidagi ma'lumotlarning elementlari bo'lib o'zgaruvchilar, o'zgarmaslar, izohlar xizmat qiladi.

O'zgaruvchilar deb - hisoblash jarayonida o'z qiymatini o'zgartiradigan kattaliklarga aytiladi. O'zgaruvchilarning nomlari (identifikatorlar)- harflardan yoki harf va raqamlardan iborat bo'ladi. Belgilar soni 8 tadan oshmasligi kerak.

O'zgarmaslar (const) deb hisoblash jarayonida o'z qiymatini o'zgartirmaydigan kattaliklarga aytiladi. Bularga ham o'zgaruvchilar

kabi nom beriladi.

Izohlar - dasturning ma'lum qismini tavsiflash uchun ishlatiladi va bu qatorda hech qanday amal bajarilmaydi, ya'ni dasturning biror blokini yaxshiroq tushunishga xizmat qiladi. Izoh (*,*) yoki {} simvollar orasida beriladi. Katta qavslar {*,*} va [*,*] lar ham ishlatilishi mumkin.

Ma'lumotlar toifalari.

Paskal tilida ma'lumotlarning toifalari ikki xil bo'ladi oddiy-skalyar va murakkab. Skalyar toifa o'z navbatida o'zgaruvchi va standart toifalarga bo'linadi. O'zgaruvchi toifaga qayd qilingan va chegaralangan toifalar kirsa, standart toifaga butun - INTEGER, haqiqiy - REAL, mantiqiy - BOOLEAN, xarakter yoki simvol - CHAR toifalar kiradi. Bundan tashqari Turbo Paskalda qator - STRING toifasi kiritilgan. Murakkab toifalarga esa ma'lumotlarning muntazam to'plam, aralash (yozuvlar), murojaat va faylli toifalari kiradi.

Ma'lumotlarning standart toifalari.

Standart turdagi ma'lumotlar dasturda maxsus xizmatchi so'zlar yordamida tavsiflanadi:

Integer - Butun toifadagi ma'lumotlar faqat butun sonlarni qabul qiladi. Ular Paskal tilida quyidagicha ifodalanadi: -7, 10, 89, -35, 0 va h.k.

Real – haqiqiy toifadagi ma'lumotlar haqiqiy sonlarni qabul qiladi va 2 xil ko'rinishda bo'ladi:

a) qo'zg'almas nuqtali haqiqiy sonlar (3.56, 0.88,-150.45)

b) qo'zg'aluvchan nuqtali haqiqiy sonlar (0.546q546E-03, 96.78*10²q96.78E02 va h.k.)

Char – belgi (simvol) toifadagi ma'lumotlar qiymat sifatida

ASCII kodlar jadvalidagi belgilarning bittasini qabul qiladi. Bu ma'lumotlar apostrof ichiga olib yoziladi. Masalan ('F', 'A', 'R', '5', '8' va h.k.)

Boolean – mantiqiy toifadagi ma'lumotlar. Bu nom ingliz matematigi Jorj Bul sharafiga shunday deb ataladi va shu turdagi o'zgaruvchilar faqat 2 ta qiymat qabul qiladi True yoki False.

Bu ma'lumotlar ustida solishtirish amallari ($>$, $<$, $=$, $<=$, $>=$), mantiqiy ko'paytirish And, mantiqiy qo'shish OR va mantiqiy inkor qilish Not amallarini bajarish mumkin. Quyidagi jadvalda Boolean toifasidagi A va B o'zgaruvchilari ustida mantiqiy amallarning bajarilishi natijasi keltirilgan:

A	B	NOT A	NOT B	A AND B	A OR B
TRUE	TRUE	FALSE	FALSE	TRUE	TRUE
TRUE	FALSE	FALSE	TRUE	FALSE	TRUE
FALSE	TRUE	TRUE	FALSE	FALSE	TRUE
FALSE	FALSE	TRUE	TRUE	FALSE	FALSE

String [N] - qator. Bu yerda N qatordagi simvollar soni (ko'rsatilmagan bo'lsa u avtomat ravishda 256 ga teng deb olinadi). Bu turdagi ma'lumotlar bitta yoki bir nechta belgilar ketma-ketligidan tashkil topib, apostrof ichida beriladi. Masalan:

'Informatika', 'Student'

Eslatma. Turbo Paskalning keyingi lahjalarida oddiy turning Byte, Word, Longint, Shortint kabi toifalari aniqlangan.

Standart funksiyalar.

Paskal tilida quyidagi standart funksiyalar aniqlangan:

Funksiyaning Paskaldagi ifodasi	Funksiyaning matematik ifodasi	Argument toifasi	Funksiya toifasi
abs(x)	$ x $	real	real
		integer	integer
sqr(x)	x^2	real	real
		integer	integer
sqrt(x)	$\sqrt{x}$	real	real
		integer	real
exp(x)	e^x	real	real
		integer	real
ln(x)	$\ln x$	real	real
		integer	real
sin(x)	$\sin x$	real	real
		integer	real
cos(x)	$\cos x$	real	real
		integer	real
Arctan(x)	$\arctg x$	real	real
		integer	real
Round(x)	x ni yaxlitlash	real	integer
Trunk(x)	xning butun qismini olish	real	integer

pred(x)	x dan oldingi qiymatni olish	integer	integer
		char boolean	char boolean
succ(x)	x dan keyingi qiymatni olish	integer	integer
		char boolean	char boolean
a div b	a ni v ga bo'lib butun qismini olish	integer	integer
		integer	
a mod b	a ni v ga bo'lib qoldig'ini olish	integer	integer
		integer	
chr(x)	x ni tartib raqamiga ko'ra simvolni aniqlash	integer	char
ord(x)	x simvolning tartib raqamini aniqlash	char	integer
odd(x)	X ning toq yoki juftligini aniqlash	integer x-toq x-juft	boolean true false

Ifodalar.

Paskal algoritmik tilida arifmetik, mantiqiy va belgili ifodalar aniqlangan.

Arifmetik ifodalar o'zgaruvchilar, o'zgarmaslar, arifmetik amal belgilari, qavs va standart funksiyalardan tashkil topadi. Arifmetik ifodada qatnashayotgan amallar ustuvorligiga qarab bajariladi. Amallar quyidagi ustuvorlikka ega.

1. Qavs ichidagi amallar va standart funksiyalarni hisoblash.
2. Ko'paytirish, bo'lish, DIV, MOD amallari.
3. Qo'shish va ayirish amali.

Arifmetik ifodada bir xil ustuvorlikka ega amallar ketma-ket kelsa, ular chapdan o'ngga qarab bajariladi.

Misol:

$$e^{\sin \sqrt{x} + \cos x^2} + tg(x + \sqrt{3}).$$

Paskalda bu ifoda quyidagi

EXP(SIN(SQRT(X))QCOS(SQR(X)))QSIN(XQSQRT(3))G'CO
S(XQSQRT(3))

ko'rinishda yoziladi

Mantiqiy ifodalar o'z ichiga mantiqiy o'zgarmaslarni, mantiqiy o'zgaruvchilarni, mantiqiy amal belgilarini, mantiqiy natija beradigan standart funksiyalarni, qavslarni hamda DIV va MOD funksiyalarini qamrab oladi. Mantiqiy ifodada ham amallar ustuvorligiga ko'ra bajariladi:

1. Qavs ichidagi amal va mantiqiy inkor amali (NOT).
2. Mantiqiy ko'paytirish (AND), DIV, MOD amallari.
3. Mantiqiy qo'shish (OR) va solishtirish amallari.

quyidagi misolda berilgan mantiqiy ifodada amallarning bajarilish ketma-ketligi keltirilgan, bu yerda Aq2, Bq4, Cq5:

1 3 2 6 5 4

(A>3) AND (AqCQ3) OR NOT(Bq5)

1)FALSE

2)FALSE

3)FALSE

4)FALSE

5)TRUE

6)TRUE

Bu mantiqiy ifodaning qiymati TRUE ga teng.

Paskal algoritmik tili dasturining tarkibiy qismi.

Dastur deb, algoritmik tilda kompyuterda masala yechish uchun qadamba-qadam yozilgan aniq va mukammal algoritmgaga aytiladi.

Paskal tilidagi dastur tarkibi 2 asosiy qismdan iborat bo'ladi. Bular dastur sarlavhasi va dastur tanasi. Dastur tanasidan keyin nuqta qo'yiladi, bu nuqta dastur oxirini bildiradi. Dastur sarlavhasi dastur tanasidan nuqta vergul bilan ajratiladi :

<dastur>::q<dastur sarlavhasi>;<dastur tanasi>

Dastur sarlavhasi PROGRAM xizmatchi so'z bilan boshlanadi, bu so'zdan keyin shu dasturga berilgan nom yoziladi (bu nom dastur ichida hech qanday maqsadda ishlatilishi mumkin emas). Dastur nomidan keyin yumaloq qavs ichida dastur parametrlari ro'yxati beriladi. Bu parametrlar fayllar nomi bo'lib, ular orqali dastur tashqi muhit bilan bog'lanadi. Mavjud bo'lgan ko'p fayllar orasidan 2ta standart matnli fayllar ajratilgan. Bular dasturni boshlang'ich ma'lumotlar va ma'lumotlarni qayta ishlash natijalari bilan

bog'laydigan INPUT va OUTPUTlardir.

Dastur sarlavhasi yozilishni quyidagi misolda ko'ramiz:

PROGRAM PASSAL (INPUT, OUTPUT);

Paskalning ko'p lahjalarida (versiya) INPUT va OUTPUT yozilmasligi ham mumkin. Umumiy holda dastur tanasi (bloki) 6 bo'limdan iborat va ular aniqlangan ketma-ketlikda joylashgan bo'lishi shart.

< blok >::q < belgilar bo'limi >

< konstantalar bo'limi >

< toifalar bo'limi >

< o'zgaruvchilar bo'limi >

< protsedura va funksiyalar bo'limi >

< operatorlar bo'limi >

Dasturning asosiy vazifasi bu kompyuterga ma'lumotlarni (berilganlarni) qayta ishlash bo'yicha buyruqlar berishdir.

Bu buyruqlar operatorlar yordamida beriladi. Shuning uchun ham operatorlar bo'limi asosiy hisoblanadi va har qanday dasturda ishtirok etishi shart.

Operatorlar bo'limidan oldingi bo'limlar ba'zi dasturlarda ishtirok etmasliklari ham mumkin. Ular tavsiflash bo'limlari deyiladi. Tavsiflashning bu bo'limlarida operatorlar bo'limida ishlatilgan kattaliklar tavsiflanadi, ya'ni e'lon qilinadi.

Belgilar bo'limi.

Dasturni ixtiyoriy operatorining oldiga belgi va ikki nuqta qo'yib qolgan operatorlardan ajratish mumkin. Operatorning belgisi bittadan ortiq bo'lishi mumkin emas. Operatorlarning belgilari bir xil bo'lishi mumkin emas. Operatorning belgisiga uning nomi sifatida qaraladi.

Dasturda ishlatilayotgan har bir belgi avval belgilar bo'limida tavsiflash yo'li bilan e'lon qilinishi kerak:

<belgilar bo'limi >::q<bo'sh>| label <belgi>{,<belgi>;

Belgilar bo'limi **label** (belgi) xizmatchi so'z bilan ochiladi, undan keyin belgilar ketma-ket vergul bilan ajratilgan konstanta shaklida yoziladi.

Belgilar bo'limida e'lon qilinayotgan belgilar ixtiyoriy ketma-ketligida yozilishi mumkin. Dasturdagi ketma-ketlikka rioya qilmasa ham bo'ladi. Belgilar (0-9999) orasidagi sonlardan olinadi (Turbo Paskalda belgilar sifatida ba'zi bir belgilarni ham ishlatish mumkin).

Metaformuladagi " I " belgisi - "yoki" va "{,}" -katta qavslar esa belgilar ro'yxat ko'rinishida berilishi mumkin degan ma'noni anglatadi.

Misol:

label 4, 5, 25;

Konstantalar-o'zgarmaslar bo'limi.

Konstanta deganda aniq bir u yoki bu turdagi qiymatni tushunamiz, dastur ishlash jarayonida bu qiymat o'zgarmaydi shu sababli o'zgarmas deb ham yuritiladi. Paskal tilida konstantalarga nom beriladi.

Konstantalar tavsiflarining hammasi konstantalar bo'limi tarkibida bo'lishi kerak.

< konstantalar bo'limi >::q<bo'sh>| sonst

<konstanta tavsifi>;<konstanta tavsifi>;

<konstanta tavsifi>::q<konstanta nomi>q

q<konstanta>(qiymat)

Konstantalar bo'limi **const** xizmatchi so'z bilan boshlanadi va nuqta vergul (;) bilan tugaydi.

Misol:

const Aq25; Bq4.5; KAFq'INFORMATIKA';

Toifalar bo'limi.

<toifalar bo'limi>::q <bo'sh> | type <toifa tavsifi> ;<toifa tavsifi>;

Avvalgi ma'ruzalardan bizga ma'lumki paskal tilida 4 standart toifadagi qiymatlar bor bular: integer (butun), real (haqiqiy), char (matnli), boolean (mantiqiy). Bu toifadagi qiymatlar bilan bir qatorda Paskal tilida boshqa toifadagi qiymatlarni ham ishlatish mumkin. Lekin standart toifalardan farqli ravishda boshqa toifalar dasturda aniq tavsiflanishi kerak. Bu tavsiflashni quyidagilarda ko'rishimiz mumkin:

<toifani tavsiflash>::q<toifa nomi>q<toifa>

<toifa>::q<toifa nomi>| <toifaning berilishi>

Bu metaformuladagi toifa nomi sifatida standart toifa nomi ham berilishi mumkin.

Misol:

TYPE

Butunqinteger;

VAR n,m: butun;

Toifalarning hamma tavsiflari toifalar bo'limida berilgan bo'lishi kerak.

Toifalar bo'limi type (tur) xizmatchi so'zi bilan boshlanadi, keyin toifalar tavsifi yoziladi. Toifalar tavsifi bir-biridan nuqta vergul (;) orqali ajratiladi.

<toifalar bo'limi>::q<bo'sh>| type <toifa tavsifi>;{<toifa tavsifi>;

Misol:

Type

```
mantiqqboolean;  
haftaq(dush,sesh,chor,pay,jum,shan,yaksh);  
ishkuniqsesh..jum;
```

Bu misolda 3ta toifa tavsiflari bor. Birinchi qatorda standart mantiqiy toifaga boshqa nom berilayapti - mantiq. Ikkinchisida yangi qayd qilingan toifa kiritilayapti va unga hafta degan nom berilayapti. Uchinchisida yangi chegaralangan toifa kiritilayapti unga ishkuni deb nom berilyapti.

O'zgaruvchilar bo'limi.

Dasturda ishlatilayotgan har bir o'zgaruvchi oldindan e'lon qilingan bo'lishi kerak. Ushbu vazifani bajarish uchun Paskal tilida <o'zgaruvchilarning tavsifi> bo'limi bor. Bu tavsifda har bir ishlatilayotgan o'zgaruvchiga nom beriladi va qabul qiladigan qiymatlar toifasi beriladi. Keyinchalik dastur bajarilish jarayonida tavsiflangan o'zgaruvchiga boshqa toifadagi qiymat berilsa bu xato deb hisoblanadi.

Alohida o'zgaruvchining tavsifi quyidagicha bo'ladi:

<o'zgaruvchi nomi>: <toifa>

Misol: x: real; y:integer;

Birinchi qatorda dastur jarayonida ishlatishga real turdagi X o'zgaruvchi kiritilayapti, ikkinchisida Y nomli butun qiymatli o'zgaruvchi kiritilayapti.

Bitta tavsif yordamida dasturda bir necha bir xil toifadagi o'zgaruvchilarni tavsiflash mumkin.

Misol:

x,y,r,h: real;

Bu tavsif ish jarayoniga 4ta-x,y,z,h nomli haqiqiy o'zgaruvchilarni kiritayapti.

O'zgaruvchilar bo'limi var (variable o'zgaruvchi) xizmatchi so'z bilan boshlanadi.

<o'zgaruvchilar bo'limi>::q<bo'sh>| var <o'zgaruvchilar tavsifi>; {,<o'zgaruvchilar tavsifi>;}

<o'zgaruvchilar tavsifi>::q<o'zgaruvchi nomi>{,<o'zgaruvchi nomi>}:<toifa>

Misol,

Var

i, j, k: interger; x,h,sum, way:real; n,m: integer; day: HAFTA;

Dasturda ishlatilayotgan o'zgaruvchilar faqat bir marotaba tavsiflanishi kerak.

Protseduralar va funksiyalar bo'limi.

Ma'lumki, dasturga ish jarayonida ixtiyoriy protsedura va funktsiyani kiritish mumkin. Demak kiritilayotgan nostandart protsedura va funktsiyalarni tavsiflash kerak.

Boshqa bo'limlardan farqli ravishda bu bo'lim maxsus xizmatchi so'z bilan belgilanmaydi. Bu bo'limning boshi Procedure yoki Function so'zi bilan boshlanadi. Protsedura tavsifi har doim funktsiya tavsifidan oldin turadi. Bu bo'lim dasturda standart protsedura va funktsiyalardan tashqari protsedura va funktsiyalar ishlatilayotgan bo'lsa va agar ularga dasturdan murojaat qilinayotgan bo'lsa ishlatiladi.

Operatorlar bo'limi.

Bu bo'lim dasturning asosiy bo'limi hisoblanadi.

<operatorlar bo'limi>::qbegin <operator>;{,<operator>} end.

Dasturni bajarish bo'limi xizmatchi so'zlar begin va end ichiga joylashgan operatorlar ketma-ketligini bajarishga keltiriladi. Operatorlar bir-biridan ";" yordamida ajratiladi.(Shuni aytish kerakki, protsedura va

funksiya bo'limidagi algoritmlar faqat operatorlar bo'limidan murojaat qilingandayoq bajariladi).

Operatorlar yordamida masalani yechish algoritmini bajarishga kerak bo'lgan amallar tavsiflanadi.

Funksional vazifalari bo'yicha Paskal tilidagi operatorlar quyidagi guruhlariga bo'linadi: o'zlashtirish, kiritish-chiqarish, boshqarish.

O'zlashtirish operatorlari - dasturda o'zgaruvchilarga ma'lum qiymatlarni o'zlashtirish uchun ishlatiladi.

Kiritish-chiqarish operatorlari - tezkor xotiraga boshlang'ich ma'lumotlarni kiritadilar va chiqaruvchi qurilmalarga hisoblashlar natijasini chiqaradilar.

Boshqarish operatorlari - dastur operatorlarini ketma-ket bajarilishini tashkil etadi(shartli o'tish, takrorlanish operatorlari).

O'z tarkibiga ko'ra operatorlar ikki turga bo'linadilar: sodda va strukturalashgan (tarkiblashgan).

Tarkibida boshqa operatorlar bo'lmagan operatorlar sodda operatorlar deyiladi.

Bir yoki bir nechta operatorlardan iborat operatorlar murakkab deyiladi.

Sodda operatorlarga o'zlashtirish, o'tish va protseduraga murojaat operatorlari kiradi. Murakkab operatorlarga shartli, tanlash, takrorlash va tarkibiy operatorlar kiradi.

Tarkibiy operator BEGIN va END xizmatchi so'zlari orasiga olingan operatorlar ketma-ketligidan iborat.

Dasturda ma'lum jarayon bitta emas, balki, bir nechta operator ketma-ketligi bilan ifodalanadi. Shu holda operator ketma-ketligi tarkibiy

operator ko'rinishida yoziladi. Tarkibiy operator ma'lum bir operatorlar ketma-ketligini operator qavslari begin (boshi) va end (oxiri) xizmatchi so'zlari orasiga olib birlashtiriladi.

<tarkibiy operator>::q begin <operator> ; {<operator>} end.

Yuqoridan ko'rinadiki, yagona (tarkibiy) operatorga keltirilayotgan operatorlar soni bir dona bo'lishi ham mumkin. Agar ularning soni ikki yoki undan ortiq bo'ladigan bo'lsa, ular nuqtali vergul bilan ajratiladi.

Aytib o'tish lozimki, paskalda nuqtali vergul operatorlarni ajratuvchi sifatida ishlatiladi, ya'ni na avvalgi va na keyingi operator tarkibiga kirmaydi. Til bo'yicha tarkibiy operator tarkibiga kiruvchi operatorlarga cheklanishlar qo'yilmaydi, ular asosiy operator yoki keltirib chiqariluvchi, shuningdek tarkibiy operatorlardan biri bo'lishi ham mumkin. Ya'ni, tarkibiy operator rekursiv xarakterga ega.

Tarkibiy operatorlarga misollar: begin i:qD end; begin y:qxG'2; x:qxQh end

begin k:q2; begin i:qD; hisobchi :qD end end

Tarkibiy operatorlarda uning tarkibiga kirgan operatorlar yozilish ketma-ketligida bajariladi. Ixtiyoriy paskal dastur tanasining operatorlar qismida hech bo'lmaganda bitta tarkibiy operator mavjud bo'ladi.

Paskal tilida oddiy algoritmlarni dasturlash asoslari.

Dasturlashtirishda bajariladigan harakatlar konsepsiyasi.

Shuni eslatib o'tish lozimki, dastur bu masalaning yechish jarayonini ifodalovchi, ma'lum bir tilda ifodalangan algoritm bo'lib, kompyuter bajaruvchi vosita hisoblanadi.

Paskal tilida yozilgan dastur to'g'ridan-to'g'ri kompyuterga emas, balki biror bajaruvchiga mo'ljallanganki, unga qanday toifadagi qiymatlar bilan ishlay olishi va qo'shimcha ko'rsatmalarsiz qanday amallarni bajara olishi belgilab berilgan bo'lishi kerak.

U yoki bu masalani yechish uchun kattaliklar ustida bajarilishi kerak bo'lgan amallarni ko'rsatish uchun algoritmik tilda operator tushunchasi xizmat qiladi.

Dasturdagi har bir operator ma'lumotlarga ishlov berishning mustaqil, mantiqan tugallangan bosqichini ifodalaydi. Paskalda operatorlarning 8 ta toifasi ko'zda tutilgan. Har qanday masalani yechish aniq qoidalar bo'yicha, berilgan kattaliklardan boshqa kattaliklarni keltirib chiqarish jarayonidan iboratdir.

Yangi kattaliklarni hosil qilish qoidasi Paskalda ifodalar orqali beriladi. Oddiy hollarda masalaning echimini bitta formula yordamida ko'rsatish mumkin.

Masalan, to'g'ri burchakli uchburchakning berilgan a va v katetlari uzunligi yordamida gipotenuza uzunligini topish uchun formula yordamida hisoblash o'tkazish kifoya. Mana shunday, berilgan ifodaga qarab, yangi qiymatni hisoblash qoidasini berish uchun paskal operatorlaridan biri- o'zlashtirish operatori ishlatiladi. Bunda berilgan formula bo'yicha hisoblangan qiymat biror o'zgaruvchiga beriladi. Dastur tuzish jarayonida faqatgina o'zlashtirish operatorlari bilan chegaralanishning iloji yo'q. Dasturda hisoblashlarning barcha yo'llari ko'zda tutilgan bo'lishi va qanday hollarda u yoki bu hisoblash yo'li tanlanishi kerakligi haqida axborot berilishi zarur.

Ko'p oddiy algoritmlarni dasturlashtirishda asosan o'zlashtirish, kiritish-chiqarish operatorlaridan foydalaniladi. Quyida shu operatorlarni

ko'rib chiqamiz.

O'zlashtirish operatori.

Masalani yechish jarayoni qator bajariluvchi bosqichlarga bo'linib ketadi. Bu bosqichlarning har birida ma'lum qiymatlar bo'yicha yangi qiymatlar hisoblanadi. Bu hisoblangan qiymatlarning ba'zilar natijaviy qiymatlar bo'lsa, ba'zilar esa oraliq qiymatlar bo'lib, keyingi bosqichlar uchun boshlang'ich qiymat bo'lib hisoblanadi.

Yangi qiymatlarni hisoblash uchun ifoda tushunchasi xizmat qiladi, har bir ifoda bitta qiymatni hisoblash qoidasini belgilaydi.

Hisoblangan qiymatni hisoblash jarayonining keyingi bosqichida foydalanish uchun eslab qolish zarur, bunday eslab qolish hisoblangan qiymatni ma'lum o'zgaruvchiga o'zlashtirish yo'li bilan amalga oshiriladi. Bunday amalni bajarish, asosiy operatorlardan biri deb, hisoblanuvchi o'zlashtirish operatori bilan bajariladi.

O'zlashtirish operatori sintaksis jihatdan quyidagicha aniqlanadi:

$\langle \text{o'zlashtirish operatori} \rangle :: q \langle \text{o'zgaruvchi} \rangle : q \langle \text{ifoda} \rangle ;$

Bu yerda 2 ta belgidan iborat bo'lgan asosiy belgi «:q» «o'zlashtirish» deb o'qiladi.

O'zlashtirish operatorining bajarilishida «:q» belgisining o'ng tomonidagi ifodaning qiymati hisoblanib, belgining chap tomonidagi o'zgaruvchi tomonidan o'zlashtiriladi.

Shunday qilib, o'zlashtirish operatori mustaqil mantiqiy tugallangan hisoblash jarayonini ifodalaydi: o'zlashtirish operatori bajarilishi natijasida ma'lum o'zgaruvchilar keyingi bosqichlarda ishlatilishi mumkin bo'lgan yangi joriy qiymatni qabul qiladi.

Turli toifadagi ifodalar aslida ko'p o'xshashliklarga ega bo'lib, ularning barchasi operandlardan, amal belgilaridan, amallarning

ixtiyoriy ketma-ketligini bajarish imkoniyatini beruvchi qavslardan tashkil topadi. Bunda operandlar uch turli: o'zgarmas, o'zgaruvchan va hisoblanadigan bo'ladilar.

O'zgarmas operand dastur tuzilayotganda ma'lum bo'lgan va bajarilishi davomida o'zgarmas bo'lgan qiymat bo'lib, boshqacha aytganda o'zgarmas operand – bu biror toifadagi konstantadir.

O'zgaruvchi operand qiymati dastur bajarilishi davomida aniqlanadi va o'zgarishi mumkin. Ammo bu operand qatnashgan ifodaning hisoblanishidan oldin uning qiymati aniqlangan bo'lishi lozim. Bunday operandlar Paskal tilining o'zgaruvchilari hisoblanadi. Sintaksis jihatdan o'zgaruvchi – qiymatni o'zlashtiradigan, shu o'zgaruvchi nomi sifatida qo'llaniladigan identifikatordir.

Hisoblanadigan operand qiymati hatto hisoblashdan avval ham aniqlanmagan bo'lib, ifoda hisoblanishi jarayonida aniqlanadi. Hisoblanadigan operandlar sifatida Paskal funksiyalarini ko'rsatish mumkin.

Operatorning o'ng tarafidagi ifoda qiymatining toifasiga ko'ra o'zlashtirish operatorini shartli ravishda bir nechta turga ajratish mumkin.

Arifmetik o'zlashtirish operatori o'zgaruvchiga arifmetik toifadagi, ya'ni REAL yoki INTEGER turdagi qiymatni berish uchun xizmat qiladi.

Bunday operatorning o'ng tarafida arifmetik ifoda, ya'ni shu turlardagi qiymatni hisoblash qoidasini beruvchi ifoda bo'lishi kerak. Agar o'zlashtirish operatorining chap qismidagi o'zgaruvchi REAL turga ega bo'lsa, arifmetik ifodaning qiymati REAL turdagi yoki INTEGER turdagi bo'lishi mumkin (bu holda olinadigan butun qiymat

avtomatik tarzda haqiqiy qiymatga aylantiriladi). Agar chap qismdagi o'zgaruvchi INTEGER turga ega bo'lsa, unda arifmetik ifoda, albatta, shu turdagi qiymatni saqlashi kerak. Arifmetik ifodaning barcha operandlari REAL yoki INTEGER turda bo'lishi kerak.

Asosiy operandlar sifatida o'zgarmaslar (ishorasiz son yoki konstanta nomi), o'zgaruvchilar va funksiyalar ishlatiladi.

Arifmetik ifodaga quyidagi misollarni (ifodaning o'ng tarafida uning hisoblanish tartibi, har bir ifoda natijasi va ishlatilgan funksiyalar qiymatlari toifalarini hisobga olgan holda) keltirish mumkin:

$$2*3Q4*5((2*3)Q(4*5)q26)$$

$$9 \text{ div } 4G'2 \quad ((9 \text{ div } 4)G'2q1.0)$$

$$40G'5G'10 \quad ((40G'5)G'10q0.8)$$

$$-\text{sqrt}(\text{sqr}(3)Q32G'2)(-\text{sqrt}(\text{sqr}(3)Q(32G'2))q-5.0)$$

$$((2Q4)G'10Q2G'4)*2 \quad (((2Q4)G'10)Q2G'4))*2q2.2)$$

Quyida arifmetik o'zlashtirish operatoriga misollar keltirilgan (bunda, x, a, b, c, r – REAL turdagi o'zgaruvchilar, pi q 3.14159 bo'lgan haqiqiy sonning nomi):

$$x:q0 \text{ (x o'zgaruvchiga nolga teng bo'lgan qiymat berilayapti);}$$

$$I:qIQ1 \text{ (I ning joriy qiymati bir birlikka orttirilayapti);}$$

$$c:q\text{sqrt}(a*aQb*b) \text{ (a va b katetlar asosida gipotenuza uzunligi hisoblanayapti);}$$

$$x:q2*pi*r \text{ (r radiusli aylana uzunligi x hisoblanayapti).}$$

Quyidagi yozuvlar sintaksis bo'yicha o'zlashtirish operatorlari emas yoki Paskalda bunday yozib bo'lmaydi:

$$3:qIQ2 \text{ (chap tomonda konstanta qo'llanishi mumkin emas);}$$

$$xq2*pi*r \text{ («q» belgi o'zlashtirish operatorining belgisi emas);}$$

$$Iq5G'4 \text{ (butun sonli o'zgaruvchiga haqiqiy qiymat berilmaydi); x:}$$

qa*-bG'2 (ketma-ket ikki amal belgisini yozish mumkin emas).

Agar o'zlashtirish operatorining chap tarafida BOOLEAN toifadagi o'zgaruvchi ko'rsatilgan bo'lsa, operatorning o'ng tarafida mantiqiy qiymat (TRUE yoki FALSE) ni hisoblovchi mantiqiy ifoda berilishi kerak.

Mantiqiy ifodada konstanta, o'zgaruvchi va funksiyalar ishlatiladi, biroq mantiqiy amalning har bir operandi BOOLEAN toifada bo'lishi kerak. Mantiqiy ifodada mantiqiy amal operandi bo'lib munosabat xizmat qiladi.

Bu tushunchani oddiylashtiraylik. Mantiqiy amal operandi
<arifmetik ifoda> <taqqoslash amali> <arifmetik ifoda>
ko'rinishidagi yozuv- munosabatdir.

Bunda <taqqoslash amali>:: $q < | < q | > | > q | q | < >$.

Agar arifmetik ifodalarning munosabati to'g'ri berilgan bo'lsa, munosabat TRUE qiymatiga, aks holda FALSE qiymatiga ega bo'ladi.

Masalan, $3 < 5$ munosabati TRUE, $3 > 5$ munosabati FALSE qiymatiga ega. Munosabatda qatnashadigan ixtiyoriy arifmetik ifoda haqiqiy yoki butun sonli bo'lishi mumkin. Butun son haqiqiy son bilan taqqoslanganda oldin haqiqiy songa o'tkaziladi, deb qaraladi.

Mantiqiy ifodaga misollar (d, b,c- mantiqiy, x, y-haqiqiy, k-butun sonli o'zgaruvchi):

T: $qx < 2 * y$; (munosabat)

H: $q \text{TRUE}$; (konstanta)

F: qd ; (o'zgaruvchi)

G: $q \text{Odd}(k)$; (funksiya)

S: $q \text{not not } d$; (inkor)

Q: $q (x > y \text{G}'2) \text{ or } d$; (mantiqiy qo'shiluvchi)

J: $qd \text{ and } (xqy) \text{ and } b$; (mantiqiy ko'paytuvchi)

N: $q (s \text{ or } d) \text{ and } (xqy) \text{ or not}$; (oddiy ifoda).

Bu yerda T,H,G,F,J,N,S,Q -BOOLEAN tursidagi o'zgaruvchilardir.

Mantiqiy ifoda qiymatini hisoblash oddiydir.

Masalan, $d \text{ or } (x*yG^2 > xQy) \text{ and not } b \text{ or } (x > 2*r)$

Ifoda qiymatini hisoblashda, bu yerda keltirilgan barcha o'zgaruvchilarning o'rniga ularning joriy qiymatlarini olish va ifodada berilgan amallarni, ularning maqomi va qo'yilgan qavslarni hisobga olgan holda bajarish kerak.

Agar o'zlashtirish operatorining chap tomonida CHAR toifadagi o'zgaruvchi ko'rsatilsa, o'ng tomonida belgili ifoda, ya'ni CHAR toifadagi qiymatli ifoda berilishi shart.

Belgili ifoda sifatida faqatgina shu toifadagi konstanta, o'zgaruvchi yoki funksiya berilishi mumkin. Belgili o'zlashtirish operatoriga misollar:

(sum, alpha, betta- CHAR toifadagi o'zgaruvchilar):

sum:q 'Q ';

alpha:qsum;

betta:qsucc(sum).

Shu narsa ma'lumki, umuman, paskal tilida arifmetik ifoda, mantiqiy ifoda va h.k. tushunchalari yo'q bo'lib, faqatgina, barcha toifadagi ifodalarni o'z ichiga oluvchi, bitta sintaksis bo'yicha aniqlanuvchi tushuncha - <ifoda>gina mavjud.

Bo'sh operator hech qanday harakatni bajarmaydigan operatoridir.

Bo'sh operatorga qoida bo'yicha operator mavjud bo'lishi kerak bo'lgan joydagi yozuvning yo'qligi to'g'ri keladi. Undan so'ng nuqtali

vergul qo'yish kerak.

Masalan:

A:qB; R:q2; ; K:q7.2;

Bu yerda uchinchi operator bo'shdir. Tarkibiy va bo'sh operatorlar shartli operatorlarda tez- tez qo'llaniladi.

Ma'lumotlarni kiritish va chiqarish.

Dasturda ma'lumotlarning qiymatlarini xotiraga kiritishni bir necha usullarda bajarish mumkin.

Sonli o'zgaruvchilarga ularning qiymatini berishda o'zlashtirish operatoridan foydalanish mumkin.

Masalan: A:q5; V:q-6.143;

Dasturni o'zgaruvchilarning turli qiymatlarida bajarish uchun READ-kiritish operatori mo'ljallangan.

Kiritish operatori quyidagicha ko'rinishlarda ishlatilishi mumkin:

1) READ(a1,a2,...,an);

bunda, a1,a2,...,an- qiymatlarini ketma- ket standart INPUT protsedura faylidan oluvchi o'zgaruvchilar. O'zgaruvchilarga qiymatlar toifasiga mos ravishda klaviaturadan kiritiladi.

Aytaylik, A, B, C o'zgaruvchilarga dastur bajarilishi davomida quyidagi qiymatlarni berish kerak bo'lsin: Aq5, Bq17, Cq6.2.

Operator READ(A,B,C) ko'rinishiga ega bo'lib, sonlar qiymatlarini dastur bajarilishi davomida quyidagicha kiritish mumkin:

5 17 6.2 [enter].

Agar o'zgaruvchi REAL toifada aniqlangan bo'lsa, uning qiymatini butun son yoki haqiqiy son ko'rinishida kiritiladi. Mashinaning o'zi butun sonni haqiqiy songa o'tkazib oladi.

Masalan,

VAR A, B:REAL;

READ(A, B) operatorining ishlatilishi natijasida 4 va 5 sonlari probel (bo'sh joy) orqali kiritish mumkin.

2) READLN; - bu operator kiritish jarayonida bo'sh qator qoldiradi;

3) READLN(a1,a2,...,an);

-operatorning bajarilishida avval a1,a2,...,an ga qiymat kiritilib, so'ng keyingi satrga o'tiladi. Bu operator oldingi ikki operatorga teng kuchlidir.

Kompyuter xotirasidagi ma'lumotlarni displey ekraniga chiqarish operatori - WRITE dir. Operator quyidagi bir nechta ko'rinishlarda ishlatilishi mumkin:

1) WRITE(a1,a2, ..., an);

- bunda a1,a2,...,an oddiy o'zgaruvchilar, o'zgarmaslar yoki ifodalar bo'lishi mumkin va ular standart OUTPUT protsedura fayliga chiqariladi. .

Masalan,

WRITE('V ning qiymati q ', V) operatori displey ekraniga:

V ning qiymati q

va undan so'ng V o'zgaruvchining qiymatini chiqaradi.

WRITE operatorida butun va haqiqiy sonlarni ma'lum formatda chiqarish mumkin. Bu format ikki nuqta orqali o'zgaruvchidan so'ng ko'rsatiladi.

Masalan,

WRITE(Y:5:2);

operatori bilan Y ning qiymatini chiqarishda, Y ning hamma

qiymatini chiqarish uchun 5 ta xona ajratilishi, ulardan ikkitasi kasr qismi uzunligini anglatadi (bunda sonning butun va kasr qismini ajratuvchi vergul (nuqta) ham hisobga olinishi zarur). Butun sonlarni chiqarishda kasr qismi formati ko'rsatilmaydi.

Aytaylik, Nq179 butun sonli qiymatni chiqarish kerak bo'lsin. Chiqarish operatori buning uchun quyidagi ko'rinishda bo'lishi mumkin:

WRITE ('Nq ', N:3)

Bu yerda sonni tasvirlash uchun 3 pozitsiya ajratilgan. Agar formatni 3 dan ortiq berilsa, masalan,

WRITE('Nq ', N:5)

unda sondan oldin ikkita bo'sh joy tashlanadi: Nq179, manfiy son uchun esa bitta bo'sh joy tashlanadi: Nq-179.

2) Paskal tilida boshqa chiqarish operatorlari ham ishlatiladi. Parametrlarsiz chiqarish operatori

- WRITELN;
- displey ekranida yangi satrga o'tishni ta'minlaydi.

3) WRITELN(a1,a2,...,an);

- chiqarish operatori oldin a1,a2,...,anlarning qiymatlarini chiqaradi, so'ng yangi qatorga o'tishni ta'minlaydi. Shunday qilib, bu ham quyidagi ikki operatorga ekvivalent

WRITE(a1,a2,...,an); WRITELN;

Masalan, A, B, C qiymatlarini kiritish uchun quyidagi lavhadan foydalanish mumkin:

WRITE('A, B, C qiymatlarini kiriting');

READ(A, B, C);

Shunday qilib, A, B, C ning qiymatlarini kiritilishidan oldin ekranga quyidagi xabar chiqariladi:

A, B, C qiymatlarini kiriting

shundan so'nggina qiymatlarni kiritish mumkin, masalan, 5 17

6.2 [Enter].

Oddiy masalalarni dasturlashga misollar.

Ba'zi-bir oddiy masalalarning dasturlarini tuzamiz.

1-misol. R radiusli sharning hajmini quyidagi formula bo'yicha hisoblansin

$$V = \frac{4}{3} \pi R^3$$

Dastur quyidagi ko'rinishda bo'ladi.

```
PROGRAM E10(INPUT, OUTPUT);
```

```
CONST
```

```
PI=3.14;
```

```
VAR
```

```
R:REAL; {Shar radiusi}
```

```
V:REAL; {Shar hajmi}
```

```
BEGIN
```

```
WRITELN('R radius qiymatini kiriting:');
```

```
READ (R);
```

```
V:=4*PI*R*R*R/3;
```

```
WRITELN;
```

```
WRITELN('Natija:');
```

```
WRITELN('Shar hajmi',V:8:3);
```

```
END.
```

2-misol. Quyidagi qarshiliklardan tashkil topgan zanjirning umumiy qarshiligini hisoblang: Ketma-ket qarshilik R_{ket} q R_1 q R_2 , parallel qarshilik R_{par} q R_1 q R_2 (R_1 q R_2) bo'lsin. R_1 ni R_1 , R_2 ni R_2 , R

ket ni R_{KET} , R_{par} ni R_{PAR} deb belgilaylik. Zanjir qarshiligini hisoblash dasturini tuzamiz:

```
PROGRAM ER(INPUT,OUTPUT);
VAR
R1,R2:REAL; {qarshiliklar}
RKET:REAL; {qetma- ket ulanish}
RPAR:REAL; {parallel ulanish}
BEGIN
WRITELN('R1 va R2 qiymatlarini kiriting:');
READ(R1,R2);
RKET:qR1QR2;
RPAR:qR1*R2G'(R1QR2);
WRITELN;
WRITELN ('Ketma-ket ulangan zanjir, Rq',RKET:8:2);
WRITELN('Parallel ulangan zanjir, Rq',RPAR:8:2)
END.
```

3-misol. CHAR toifadagi o'zgaruvchilar bilan PRED, SUCC, ORD va CHR funksiyalari ishlatilgan dastur:

```
Program belgi(input,output);
Var x,x1,x2,x3:char;
n:integer;
Begin
x1:q 'l';
writeln (x1);
x2:qpred(x1); writeln('predq ',x2);
x3:qsucc(x1); writeln('succq ',x3);
x:q 'A';      writeln(x); n:qord(x);
```

```
x:qchr(n);    writeln(x);  
end.
```

4-misol. Mantiqiy toifa qatnashgan dasturga misol:

```
Program mantiq(input,output);  
Var x:integer; mant:boolean;  
Begin readln(x);  
    mant:qx>3;  
    writeln(mant)  
end.
```

Tilning boshqaruvchi konstruksiyasi. Tarmoqlanuvchi hisoblash jarayonlarini algoritmlash va dasturlash.

Ko'pgina masalalarni yechishda ba'zi bir jarayonlar ma'lum shart yoki shartlarning qo'yilishiga nisbatan bajariladi. Bunday jarayonlar tarmoqlanuvchi jarayonlar deb yuritiladi va bu jarayonlarning algoritmik tavsiflari bilan avvalgi boblarda tanishgan edik.

Tarmoqlanuvchi hisoblash jarayonlari oddiy va murakkab bo'lishi mumkin. Bu esa jarayondagi tarmoqlar soniga bog'liq. Ma'lum bir tarmoqlanuvchi jarayon tarkibida yana tarmoqlanishlar bo'lishi mumkin. Bunday tarmoqlanishlari bor bo'lgan hisoblash jarayonlari murakkab tarmoqlanuvchi hisoblash jarayonlari deb ataladi.

Paskal tilida tarmoqlanuvchi jarayonlarni dasturlash.

Paskal tilida tarmoqlanuvchi jarayonlarni dasturlash uchun shartsiz, shartli o'tish va tanlash operatorlaridan foydalaniladi.

Shartsiz o'tish operatori

Dasturda ba'zi bir hollarda boshqaruvni to'g'ridan-to'g'ri biron bir operatorga uzatishga, ya'ni dasturning bajarilish ketma-ketligini

buzishga to'g'ri keladi. Bu jarayon shartsiz o'tish operatori yordamida bajariladi.

Shartsiz o'tish operatorining umumiy ko'rinishi quyidagicha:

GOTO < operator belgisi>;

Bu yerda operator belgisi boshqaruv uzatiladigan operator belgisidir. Belgi sifatida 0-9999 oraliqdagi natural sonlar va CHAR toifasidagi belgilar (simvollar) ishlatiladi.

Belgi tavsiflash bo'limining LABEL bo'limida albatta tavsiflangan bo'lishi shart.

Misol:

GOTO 25;

.

.

25: y:qx*x;

Belgili operatorlarda belgi bilan operator o'rtasida « : » belgisi qo'yiladi. Bu operatorning noto'g'ri qo'llanilishi dasturning bajarilishiga halaqit beradi. Shuning uchun, dasturda bu operatorning kamroq uchrashi maqsadga muvofiqdir.

Shartli o'tish operatori

Dasturda boshqaruvni ma'lum shart asosida u yoki bu tarmoqqa uzatish shartli o'tish operatori yordamida amalga oshiriladi. Shartli o'tish operatori ikki xil ko'rinishda ishlatilishi mumkin: to'liq va qisqa.

Shartli o'tish operatorining to'liq ko'rinishini ko'rib chiqamiz. Uning metaformulasi quyidagicha yoziladi:

<to'liq shartli operator>::q IF <mantiqiy ifoda> THEN

<operator> ELSE <operator>;

ya'ni

IF<mantiqiy ifoda>THEN S1 ELSE S2;

Bu yerda IF (agar), THEN (u holda) va ELSE (aks holda) degan xizmatchi so'zlar, S1 va S2 ixtiyoriy operatorlar.

Operatordagi mantiqiy ifoda boshqaruvni uzatish shartini belgilaydi.

Operatorning ishlash tartibi quyidagicha: Agar keltirilgan mantiqiy ifoda TRUE (rost) qiymatni qabul qilsa, ya'ni qo'yilgan shart bajarilsa, THEN - xizmatchi so'zidan keyingi operator bajariladi, aks holda ELSE xizmatchi so'zidan keyingi operator bajariladi.

Mantiqiy ifodalarda munosabat amallari, mantiqiy amallar ishlatilishi mumkin. Masalan,

$A > B$, $A \neq B$, $X < 4.55$, $2QZ > 0$, $XQY < q1$ va h.k.

Shartlar oddiy va murakkab bo'lishi mumkin.

Agar mantiqiy ifodada bitta munosabat amali berilgan bo'lsa, "oddiy shart" ni ifodalaydi.

Paskal tilida quyidagi munosabat amallaridan foydalaniladi:

Munosabat amal belgisi			
q	Teng	$2q2$;	XqU
$< >$	Teng emas	$2<>3$;	$X<>U$
$<$	Kichik	$2<3$;	$X<U$
$>$	Katta	$5>4$;	$X>U$
$<q$	Katta emas	$1<qZ$;	$X<qU$
$>q$	Kichik emas	$1>qZ$;	$X>qU$

Kattaliklar orasidagi shartlar HAM, YoKI, EMAS (Paskal tilida AND, OR, NOT) mantiq amallari belgilari orqali bog'lanuvchi bir necha munosabatlardan iborat bo'lsa "murakkab shartlar" deb ataladi.

Masalan,

Matematik yozilish Algoritmik tilda yozilishi

1) $6 < qX < 10$ $(X > q6) \text{ AND } (X < 10)$

2) $aqbq0$ $(aq0) \text{ AND } (bq0)$

3) $1 < X < q4$ $(X > 1) \text{ AND } (X < q4)$

AND amalining natijasi uning ikkala argumenti ham rost

bo'lsagina rost bo'ladi.

OR amalining natijasi rost bo'lishi uchun argumentardan birining rost bo'lishi yetarli.

NOT amalining natijasi argumentning inkor qiymatiga teng, ya'ni argument rost bo'lsa - natija yolg'on, argument yolg'on bo'lsa - natija rost bo'ladi. Masalan:

$(4 < 5) \text{ AND } (5 < 100)$ - mantiqiy ifoda TRUE (rost),

$(\text{SIN}(X) > 1) \text{ AND } (5 \text{ div } 2q0)$ ifoda FALSE (yolg'on) qiymatga teng. Shuni ta'kidlab o'tish kerakki, agar mantiqiy ifodalar, biz yuqorida aytganimizdek, mantiqiy amallar yordamida (AND, OR, NOT) murakkab ko'rinishga ega bo'lsa, ular qavslarga olib yoziladi.

Shartli o'tish operatorining ishlatilishini misollarda ko'rib chiqamiz.

1) IF $U > 0$ THEN D:qSQRT(Y) ELSE D:qU;

Shartli operatorida $U > 0$ bo'lsa D: q SQRT(y) operatori, aks holda D: qU operatori bajariladi.

2) IF $(X \bmod 2 \neq 0) \text{ AND } (X > 0)$ THEN X:qSQRT(X) ELSE X:qSQR(X);

Operatorida X ning qiymati juft va musbat bo'lsa, uning qiymati ildiz ostidan chiqariladi, aks holda kvadratga oshiriladi.

Ayrim algoritmlarda ba'zan shunday hol uchrashi mumkinki, bunda hisoblash jarayonida ayrim amallar ba'zi bir shartlar bajarilgandagina hisoblanadi, aks holda, hech qanday amal bajarilmaydi. Bu holda shartli o'tish operatorini qisqa ko'rinishda ifodalash mumkin. Uning metaformulasi yozilishi quyidagicha:

<qisqa shartli operator> :: q IF <mantiqiy ifoda> THEN <operator> ,

ya'ni

IF <mantiqiy ifoda> THEN <operator>;

Operatorning bajarilish tartibi quyidagicha: agar mantiqiy ifoda TRUE (rost) qiymat qabul qilsa, operator bajariladi, aks holda IF dan keyingi turgan operator bajariladi.

Misol:

IF $X < 0$ THEN T:qX*X;

Shartli o'tish operatorining metaformulasidagi operator o'rnida o'z navbatida yana shartli o'tish operatorining to'la va qisqa ko'rinishlari ishlatilishi mumkin. Masalan,

1) IF B1 THEN IF B2 THEN A;

Bu yerda B1,B2 -mantiqiy ifoda, A - operator.

Bu operatorning bajarilishi natijasida V1 mantiqiy ifoda

tekshiriladi, agar TRUE qiymat qabul qilsa V2 mantiqiy ifoda tekshiriladi, u ham rost (TRUE) bo'lsa A operator bajariladi. Agar mantiqiy ifodalar V1 yoki B2 yolg'on bo'lsa (FALSE) shartli o'tish operatoridan keyingi operator bajariladi.

```
2)      IF B1 THEN A1
        ELSE IF B2 THEN A2
        ELSE A3;
```

bu yerda B1,B2 - mantiqiy ifoda, A1,A2,A3 - operatorlar.

Operatorlarda birinchi ELSE dan keyingi operator o'rnida yana to'liq IF operatori ishlatilgan.

Misol:

```
IF X <= A THEN Z:=SIN(X)
        ELSE IF X > B
        THEN Z:=SIN(X)*COS(X)
        ELSE Z:=COS(X);
```

Agar birinchi shart bajarilsa Z: SIN(X) operatori, aks holda ikkinchi shart $X > B$ tekshiriladi va bu shart bajarilsa Z: SIN(X)*COS(X) hisoblanadi, aks holda Z: COS(X) hisoblanadi.

Agar shartli o'tish operatorida THEN yoki ELSE dan keyin bir nechta operator guruhi bajarilsa ular tarkibiy operator ko'rinishida yozilishi kerak, ya'ni operatorlar qavsi - BEGIN va END lar orasida yoziladi.

Misol:

Agar $A < 0$ bo'lsa, $X \leq 5$, $Y \leq 23$ aks holda X va Y?0 ni o'zlashtirsin, u holda IF operatori quyidagicha yoziladi.

```
IF A<0 THEN BEGIN X:=5; Y:=23 END
        ELSE BEGIN X:=0; Y:=0 END;
```

Tanlash operatori

Juda ko'p tarmoqlanish jarayonlarida tarmoqlanish 2ta yoki undan ortiq tarmoqqa ajraladi. Umuman olganda buni bizga tanish shartli o'tish operatori yordamida amalga oshirish mumkin:

```
IF B1 THEN A1 ELSE
IF B2 THEN A2 ELSE
.....
IF BK THEN AK ;
```

Lekin bu hollarda shartli o'tish operatorlarining yozilishi noqulay.

Ko'p hollarda dasturchi uchun shartli operatorning umumiydashgan ko'rinishi — tanlash (variant) operatorini ishlatish qulay. Tanlash operatorining metaformulasi quyidagicha yoziladi:

<tanlash operatori>::q CASE <operator selektori> OF <tanlash ro'yxati elementi> ;

{,<tanlash ro'yxati elementi >} END,

bunda:

<operator selektori>::q <ifoda>,

<tanlash ro'yxati elementi>::q <tanlash belgilarining ro'yxati>:<operator>

<tanlash belgilarining ro'yxati>::q<tanlash belgisi>, {,<tanlash belgisi>}

<tanlash belgisi>::q <o'zgarmas >.

Tanlash operatorining umumiy ko'rinishi:

SASE C OF

M1 : A1;

M2 : A2;

.

.

.

Mp : Ap

END;

Bu yerda CASE (tanlash) -xizmatchi so'z, OF (dan), S - selektor, Mi - operatorlar belgilari, Ai -operatorlar (Iq1 dan n gacha).

SASE operatori tarmoqlanish jarayonini berilgan bir nechta

operatoridan birini tanlash yo'li bilan amalga oshiradi. Tanlash operatorida barcha operatorlar, shu jumladan bajarilishi uchun tanlangan operator ham aniq ravishda keltiriladi (berilgan operatorlar ketma-ketligi chegaralangan). Bajarilishi kerak bo'lgan operator yoki operatorlar ketma-ketligi operator selektorining qiymatiga ko'ra aniqlanadi.

Operator selektori sifatida haqiqiy bo'lmagan, skalyar ko'rinishdagi har qanday ifoda yoki o'zgaruvchi ishlatilishi mumkin. Operatorning ishlashida uning tarkibidagi har bir operator tanlash belgisi deb ataluvchi belgi bilan ta'minlanadi. Bu belgi operatorning bajarilishi uchun zarur bo'lgan selektorning maxsus qiymatini qabul qiladigan selektorning tavsifiga mos konstantadir. Operator bir nechta mavjud qiymatlar bilan ishlashi uchun, unda tanlash belgilari ro'yxati keltirilishi kerak.

Operator bajarilishida dastlab selektorning qiymati hisoblanadi. So'ngra selektorning qiymatiga mos belgili operator bajariladi. Agar operatorlar ketma-ketligida bunday belgili operator topilmasa, dasturda xato qayd etiladi. Shuning uchun dastur bajarilishi jarayonida selektorning qiymatiga mos keladigan maxsus belgili operator operatorlar ketma-ketligida bo'lishi shart. Bunda tanlash operatorida beriladigan belgilar belgilarni tavsiflash bo'limida keltirilmaydi.

Tanlash operatoridagi belgili operatorlar oddiy belgiga ham ega bo'lishlari mumkin. Bu holda oldin tanlash belgilari, so'ngra oddiy belgilar yoziladi. Shuni ham inobatga olish lozimki, tanlash operatoriga faqat CASE xizmatchi so'z orqali kirish mumkin, ya'ni tanlash operatoridan tashqaridagi o'tish operatori orqali bu operatorga murojaat qilish mumkin emas. Tanlash operatorining bajarilishi uning tarkibidagi operatorlar ketma-ketligidagi bitta operatorning bajarilishiga olib keladi.

Shuning uchun ularning biridan biriga GOTO operatori yordamida o'tish xato bo'ladi.

Shartli o'tish operatorining quyidagi ko'rinishi

IF B THEN A1 ELSE A2

tanlash operatorining quyidagi qurilishiga ekvivalentdir:

SASE V OF

TRUE: A1;

FALSE:A2;

END;

qisqa ko'rinishdagi shartli o'tish operatorining quyidagi IF B THEN A ko'rinishi tanlash operatorining quyidagi ko'rinishiga ekvivalentdir.

SASE V OF

TRUE: A;

FALSE:

END;

Misol:

SASE T OF

'*','G': R: q1;

'Q','-': R:q2

END;

Bu operatorning bajarilishi natijasida, agar T-belgili o'zgaruvchi "Q" yoki "-" belgi qiymatlarni qabul qilsa R o'zgaruvchi 2 qiymatni, agar T - o'zgaruvchi "*" yoki "G" belgini qabul qilsa, R o'zgaruvchi 1 qiymatni qabul qiladi.

Misol:

$ax^2QbxQcq0$ kvadrat tenglamaning ildizlarini topish dasturi tuzilsin.

Program Ildiz;

Label 20;

var A,B,C,D,E,F,X,X1,X2,Z:Real;

Begin

Read (A,B,C);

If Aq0 Then Begin X:q- BG'C; Writeln (x); Goto 20 End

Else Begin D:qB*B-4.0*A*C;

Z:q2.0*A;

E:q-BG'Z;

F:qsqrt(ABS(d))G'Z end;

if D>q0 Then Begin X1:qEQF;

```

X2:qE-F;
Writeln (x1,x2); end;
Else If Dq0 Then Begin X:qE; writeln(x) End
Else Writeln ('echim yo'q');
20: End.

```

Takrorlanish jarayonlarini tashkil qilish. Takrorlanish operatorlari

Shu paytgacha ko'rib chiqilgan operatorlarning hammasi aniq bir marotaba bajariladigan buyruqlar tizimini tashkil qiladi. Ko'rib chiqilgan operatorlar orqali faqatgina oddiy hisoblashlarnigina bajarish mumkin. Haqiqatan bunday oddiy ko'rinishdagi masalalarga dastur tuzib, uni kompyuterga kirgazib, xatosi bo'lsa uni to'g'rilab natija olishga sarflangan vaqtni yarmisini sarflagan holda, bunday masalalarni oddiy kalkulyatorda ham, dastur tuzmasdan hisoblash mumkin.

Amaliyotda murakkab jarayonlarni dasturlashda ma'lum buyruqlar ketma-ketligini ma'lum shartlar asosida qayta-qayta bajarish zarurati tug'iladi. Ma'lum bir o'zgaruvchining turli qiymatlarida ma'lum buyruqlar tizimining biron bir qonuniyatga asosan qayta-qayta bajarilishi takrorlanuvchi hisoblash jarayoni (sikl) deb ataladi.

Takrorlanuvchi hisoblash jarayonining takror-takror hisoblanadigan qismini takrorlanishning tanasi (jismi) deb ataladi.

Takrorlanish ichida qiymatlari o'zgarib boradigan o'zgaruvchini takrorlanish o'zgaruvchisi yoki takrorlanishni boshqaruvchi o'zgaruvchisi (sikl parametri) deb yuritiladi.

Takrorlanuvchi jarayonning algoritmi umumiy holda quyidagilarni o'z ichiga olishi kerak:

1. Takrorlanishni tayyorlash - takrorlanishni boshlashdan oldin takrorlanishda qatnashadigan o'zgaruvchilarning boshlang'ich

qiymatlari yoki takrorlanish o'zgaruvchisining boshlang'ich qiymati o'rnatiladi, takrorlanish o'zgaruvchisining o'zgarish qadami belgilanadi.

2. Takrorlanish tanasi - takrorlanish o'zgaruvchilarining turli qiymatlari uchun takror bajariladigan amallar ketma-ketligi ko'rsatiladi.

3. Takrorlanish o'zgaruvchisiga yangi qiymat berish - har bir takrorlanishdan avval o'zgaruvchiga o'zgarish qadamiga mos ravishda yangi qiymat beriladi.

4. Takrorlanishni boshqarish - takrorlanishni davom ettirish sharti tekshiriladi, takrorlanishning boshiga o'tish ko'rsatiladi.

Paskal algoritmik tilida uch xil ko'rinishda takrorlanuvchi hisoblash jarayonini tashkil qilish mumkin va bu jarayonlarni dasturlash uchun maxsus operatorlar belgilangan:

-avval sharti tekshiriladigan (sharti oldin kelgan"toki") takrorlanish jarayoni, bu maxsus WHILE operatori orqali amalga oshiriladi;

- sharti keyin tekshiriladigan ("...gacha") takrorlanish jarayoni, bu jarayonni maxsus REPEAT operatori orqali amalga oshiriladi;

- parametrli takrorlanish jarayoni, bu jarayon maxsus FOR operatori yordamida amalga oshiriladi.

Yuqorida keltirilgan jarayonlar ustida alohida-alohida mavzu sifatida to'xtalib o'tish lozim bo'ladi.

Takrorlash uchun savollar

1. Chiziqli jarayon deb nimaga aytiladi?
2. O'zlashtirish operatori (o'zgaruvchi, operand, o'zlashtirish belgisi).
3. Paskal tilida toifalar tushunchasi (standart, skalyar,

murakkab, oddiy, o'zgaruvchi).

4. Kattaliklarni kiritishni tashkil qilish (const, o'zlashtirish, kiritish operatori).

5. Kattaliklarni bosmaga chiqarishni tashkil qilish (chiqarish operatori).

6. Paskal tilida dastur tuzish (dastur, blok).

Fizika fanidan ba'zi formulalarni hisoblashda

“Turbo Paskal” dasturlash tilidan foydalanish usullari

Bugungacha ishlab chiqarilgan dasturlash tillaridan keng tarqalgani **Paskal** (Pascal) dasturlash tilidir. Paskal dasturlash tili 1969-yil Nikalas Virt tomonidan ishlab chiqildi. Paskal tili dasturlashni o'rgatish maqsadida ishlab chiqilgan bo'lsada yuqori malakali dasturchilar orasida keng tarqaldi. Albatta samarali dasturlash tillari o'zgarishsiz qolmaydi. Shu sababli ham turli rusumdagi kompyuterlar uchun Paskal tilining ularga moslashtirilgan naqlari ishlab chiqilgan bo'lib, ular Paskal tilining boshlang'ich naqlidan farq qilishi mumkin.

Paskal dasturlash tilining keng tarqalishi va qo'llanilishiga asosiy sabab — dasturning soddaligi va undan foydalanishning qulayligidir. Dastavval Paskal tili universitetlarda qo'llanilgan bo'lib, keyinchalik uning turli rusumli kompyuterlar uchun naqlari ishlab chiqildi. Hozirgi kunda umumta'lim maktablarida **“Turbo Paskal”** dasturlash tillari o'qitilib kelmoqda.

1981 - yilda Paskal tilining xalqaro standarti taklif etildi. Paskal tilining Borland firmasi tomonidan ishlab chiqilgan **Turbo Paskal 7.0** naqlidan hozirgi davrda keng foydalaniladi. U foydalanuvchilar uchun juda qulay tizim - dasturlashning integrallashgan muhitiga ega.

NAZARIY MA'LUMOTLAR:

Sodda dastur quyidagicha tuzilishga ega: o'zgaruvchilarni tavsiflash (**Var**), dastur boshlanishi (**Begin - boshlash**), kiritish-chiqarish operatorlari (**Read, ReadLn** – o'qimoq, **Write, WriteLn** – yozmoq), dasturni yakunlash uchun **End** va nuqta qo'yiladi..

DASTURLASH UCHUN MASALALAR:

1-topshiriq. *Zichlikni hisoblash dasturini tuzing:*

```
Program Zichlik;  
Var p,m,V: Real;  
Begin  
WriteLn ('Massani kiriting:');ReadLn(m);  
WriteLn ('Hajmni kiriting:');ReadLn(V);  
P:=m/V;  
WriteLn('p= ', 'kg/metrikub.');
```

End.

1. Dasturni kompyuterga kiritib (**Ctrl+F9**) so'ngra (Alt+F5) tugmalarini birgalikda bosiladi.
2. Massa va hajm uchun qiymat so'raladi.
3. Formula bo'yicha qiymatlar hisoblanib natija chiqariladi.

2-topshiriq. *Kuchni hisoblash dasturini tuzing:*

```
Program Kuch;  
Var F,m,a: Integer;  
Begin  
WriteLn ('Massani kiriting:');ReadLn(m);  
WriteLn ('Tezlanishni kiriting:');ReadLn(a);  
F:=m*a;  
WriteLn('F= ', 'kg/metrsekundkvadrat.');
```

End.

1. Dasturni kompyuterga kiritib (**Ctrl+F9**) so'ngra (Alt+F5) tugmalarini birgalikda bosiladi.

2. Massa va tezlanish uchun qiymat so'raladi.
3. Formula bo'yica qiymatlar hisoblanib natija chiqariladi.

Mustaqil ishlash uchun topshiriqlar.

1. Tosh yo'lining bosh qismida avtotransport uchun 1-rasmdagi belgi qo'uyilgan. Bu yo'l harakat belgisi bo'lib, qanday fizik kattalikni bildiradi?

Tezlikni. Km/soat birligida belgilanadi.

2. Odam 18 km masofani 3 soatda bosib o'tgan bo'lsa, u qanday tezlik bilan harakatlangan?

3. Avtomobil 54 km/soat tezlik bilan harakatlanmoka. Bu tezlikni m/s da ifodalang.

4. Paxta omboridagi transportyor tasmasining tezligi 0,18 m/s. Tasmaning uzunligi 32,4 m bo'lsa, transportyor paxtani qancha vaqtda tepaga olib chiqadi?

5. 15 m/s tezlik bilan harakatlanayotgan velosipedchi 9 km yulni qancha vaqtda bosib o'tadi?

6. Yo'lovchi poezdi «Toshkent-Sirdaryo» orasidagi 81 km masofani 90 minutda bosib o'tgai. Poezdning o'rtacha tezligini km/soat hisobida ifodalang.

7. Toshkent metro poezdi «Chilonzor - Ipak yo'li» bekatlari orasidagi 24 km masofani 20 minutda bosib o'tdi. Poezdning o'rtacha tezligini aniqlang.

8. Ikkita otlikdan biri 45 s davomida 4 m/s tezlik bilan harakatlangan, ikkinchisi yo'lining shu qismini 50 s da bosib o'tgan. Ikkinchi otliqning o'rtacha tezligini toping.

9. O'quvchi maktabdan uygacha 1000 qadam sanagan. Bu masofani u 6 minutda bosib o'tgan. Agar har bir qadamning uzunligi taxminan 60 sm bo'lsa, o'quvchi uyda maktabgacha bo'lgan masofani qanday o'rtacha tezlik bilan o'tgan? O'rtacha tezlikni m/min da ifodalang.

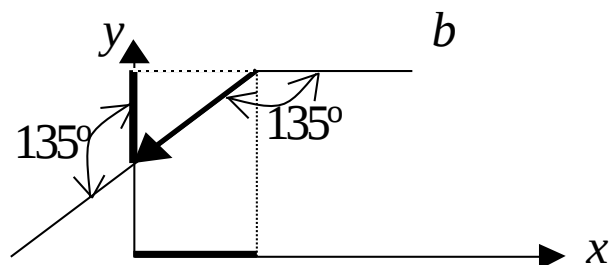
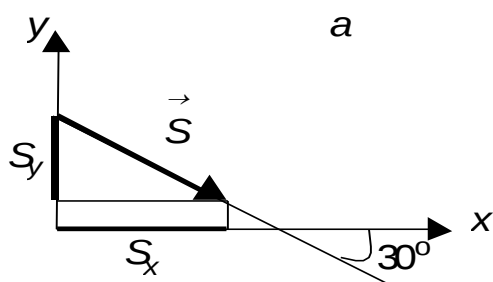
10. Toshkent metropolitenining poezdi «Sobir Raximov» va «Hamza» bekatlari orasidagi 6 km yulni 8 minutda bosib o`tdi. Butui yo`ldagi o`rtacha tezlikni (m/s va km/soat hisobida) toping.

11. Yerning sun'iy yo`ldoshi 12 sekundda 96 km masofani bosib o`tdi. Yo`ldosh qanday tezlikda uchgan.

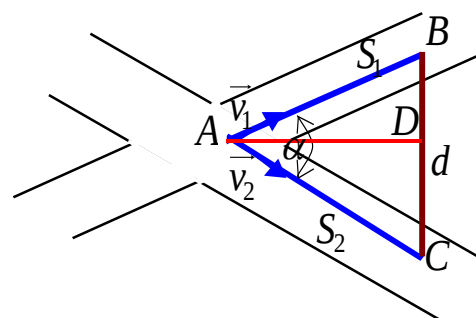
12. Traktor 3,6 km/soat tezlik bilan uzunligi 120 m bo`lgan, egatni yumshatmoqda. Traktor egatning boshidan oxirigacha qancha vaqtda yetib boradi.

13. Avtomobil uzunligi 109 km bo`lgan aylanma yo`ldan ikki marta o`tdi. Avtomobil bosib o`tgan yo`l va uning ko`chishi nimaga teng?

14. Ko`chish vektorining moduli  $S=10$  km. 14 a, b rasmda ko`rsatilgandek joylashgan vektorning koordinata o`qlaridagi proeksiyasini aniqlang.



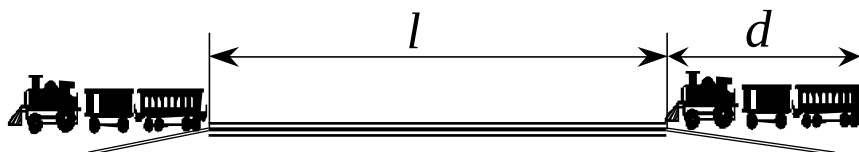
15. Bir-birini 60° burchak ostida kesib o`tuvchi yo`llarda ikkita avtomobil bir xil $72 \frac{\text{KM}}{\text{coat}}$ tezlik bilan tekis harakatlanmoqda. Avtomobillar chorraha-da uchrashganlaridan so`ng qancha vaqtdan keyin ular orasidagi



masofa 3 km ga teng bo'ladi?

Berilgan: $\alpha = 60^\circ$, $|\vec{v}_1| = |\vec{v}_2| = 72 \frac{\text{KM}}{\text{coat}} = 20 \frac{\text{M}}{\text{c}}$, $d = 3\text{KM} = 3 \cdot 10^3 \text{ M}$, $t - ?$

16. Uzunligi 120 m bo'lgan poezd ko'priklarni bo'ylab $180 \frac{\text{KM}}{\text{coat}}$ tezlik bilan tekis harakatlanmoqda. Agar ko'priklarning uzunligi 480 m bo'lsa poezd qancha vaqt ichida ko'priklarni bosib o'tadi? Bu yerda poezdni moddiy nuqta deb qarash mumkinmi?



17. Avtomobil $72 \frac{\text{KM}}{\text{coat}}$ tezlik bilan harakatlanmoqda. Haydovchi chorrahada sariq chiroqni ko'rib tormoz tepkisini bosdi. Avtomobil 5 s dan so'ng to'xtaydi. Avtomobilning tormoz yo'lini va uning harakatlanish tezlanishini hisoblang.

18. Massasi 100 kg bo'lgan odam chang'ida tepalikdan tushayotgan yo'lning gorizontal qismi bo'ylab 20 m masofani bosib o'tdi. Chang'ining qorga ishqalanish koeffitsienti 0,02 bo'lsa, yo'lning shu qismida ishqalanish kuchining ishini toping.

19. Kosmik kemada velosiped nasosi bilan veloseped shinasiga dam berish mumkinmi?

20. Agar qo'zg'almas qayiqda turgan 80 kg massali ovchi gorizontal yo'nalishda o'q uzsa, massasi 120 kg bo'lgan qayiq qanday tezlik oladi? Pitra o'qning massasi 20 g, uning boshlang'ich tezligi $400 \frac{\text{M}}{\text{c}}$.

Xulosa

Zamonaviy dasturlash tillari kompyuterning ichki mashina tilidan keskin farq qiladi va kompyuter bevosita ana shu tilda ishlay olmaydi. Buning uchun dasturlash tilidan kompyuter tushunadigan tilga tarjima qiluvchi maxsus dastur - translatordan foydalaniladi. Dasturni translyatsiya qilish va bajarish jarayonlari vaqtlarga ajraladi.

O'zining soddaligi, mantiqiyligi va samaraliligi tufayli bu til butun dunyoga tezda tarqaldi. Hozirgi paytda barcha hisoblash mashinalari, xususan, mikrokompyuterlar ham shu tilda ishlash imkoniyatiga ega. Dasturlar matnining to'g'riligini osonlik bilan tekshirish mumkinligi, ularning ma'nosi yaqqol ko'zga tashlanishi va oddiyligi bilan ajralib turadi. Paskal tili ancha murakkab va ko'p vaqt oladigan hisoblash ishlarini bajarishga mo'ljallangan tarkiblashtirilgan dasturlar tuzishga imkon beradi. Paskal tilining yana bir afzalligi shundan iboratki, foydalanuvchi xatolikka yo'l qo'ymasligi uchun yoki xato yozib qo'ygan bo'lsa, tez tuzatib olishi uchun dasturda ishlatilgan o'zgaruvchilar oldindan qaysi turga (toifaga) mansub ekanligi belgilab qo'yilgan bo'ladi. Shu bilan birga dasturning barcha elementlari haqida ma'lumot tavsiflash bo'limida mujassamlashgan bo'ladi. Operatorlar soni esa minimal darajada kamaytirilganidir.

Bundan tashqari fizikaga doir bo'lgan misol va masalalarni ham paskal tilida dasturini tuzib undan natijalar olish mumkin. Qulaylik tomoni shundaki, bitta tuzilgan dastur yordamida bir necha misol va masalalarni natijalarini olish mumkin. Yuqorida fizikaga oid bo'lgan misollar namuna sifatida dasturlari tuzilgan va mustaqil ishlashlari uchun topshiriqlar ham berilgan.

Foydalanilgan adabiyotlar

1. Aripov M.M, Irmuxamedova R.M va boshqalar. Informatika. Axborot texnologiyalari, o'quv qo'llanmasi, 1-2-qism. Toshkent 2005
2. G'ulomov S.S., Begalov B., Shermuhamedov A. Axborot tizimlari va texnologiyalari. Toshkent: «Sharq»- 2000.
3. Абрамов В.Г. Трифонов Н П. Трифонова Г.Н. Введение в язык Паскаль. Ўқув қўлланма – М: Наука, 1988.
4. Aripov M. Informatika va hisoblash texnikasi asoslari. Toshkent . "Universitet" 2001. 360 b.
5. Персон Р, Роуз. К “Word- 97 в подлиннике” 1999. BHV- С. Петербург. 1090 с.
6. Симонович С., Евсеев Г., Алексеев А., Специальная информатика. Москва. АСТ пресс. 1999. 480 с.
7. Sagatov M.V., Yakubov A.X., Irmuxamedova R.M. va boshqalar. Informatika. Axborot texnologiyalari (Ma'ruzalar matni), 2-nashri, ToshDTU, T.: 2002. 196 b.
8. Turdiyev N. 6-Fizika. Toshkent. “O'qituvchi”- 2010.
9. Habibullayev P.Q va boshqalar. 7-Fizika. Toshkent. “O'qituvchi”2010
10. Habibullayev P.Q va boshqalar. 8-Fizika. Toshkent. “O'qituvchi”2010
11. Habibullayev P.Q va boshqalar. 9-Fizika. Toshkent. “O'qituvchi”2010

