**№1-ma’ruza. Ma’lumotlar tushunchasi. Ma’lumotlarni ifodalash bosqichlari.**

**Reja:**

1. Asosiy tushuncha va ta’riflar.
2. Ma’lumotlarni tasvirlash bosqichlari.
3. Ma’lumotlar tuzilmasini klassifikatsiya qilish.
4. Ma’ lumotlarni abstrakt toifalari.

***Kalit so‘zlar:*** *butun toifa, haqiqiy toifa, mantiqiy toifa, belgili toifa, ko‘rsatkichli toifa, sanaladigan toifa, ma’lumotlar tuzilmasi, ma’lumotlar toifalari.*

1. **Asosiy tushuncha va ta’riflar**

*Ma’lumotlar tuzilmasi* - bu xotirada tashkil etiladigan elementlar yig‘indisi bo‘lib, ular ustida dastur yordamida amallar bajariladi.

*Ma’lumotlar tuzilmasi* - bu bironta toifaga tegishli bo‘lgan va o‘zaro ma’lum munosabatga ega bo‘lgan elementlar to‘plamiga aytiladi.

*Ma’lumot* - bironta qiymat yoki qiymatlar to‘plami hisoblanadi. Misol uchun bu bironta eksperiment natijalari, yoki talabalarning imtixon ballari bo‘lishi mumkin.

*Ma’lumotlar tuzilmasi elementi* - bu qiymatlar to‘plamining bir bo‘lagi hisoblanadi. Tuzilma elementi - qiymatlar jamlanmasi bo‘lib, misol uchun

talabalarning ismi, sharifi, yoshi har bir fandan olgan baxosi va x.k. larni keltirish mumkin. Elementlar 2 ga bo‘linishi mumkin:

-Element sifatida ma’lumotlar guruhi olib qaraladi. Bunda elementlar yana qism bo‘laklarga bo‘linishi mumkin. Masalan, ota-onalar maydoni talabalarning ota va onalari xaqida ma’lumot saqlaydigan alohida maydonlardan tashkil topadi.

-Elementar, ya’ni bo‘linmas, bunda element qism bo‘laklarga ajratilmaydi.

*Obyekt* - bu xususiyatlar va attributlariga ega bo‘lgan va bu xususiyatlarga qiymat qabul qilishi mumkin bo‘lgan tuzilma hisoblanadi. Masalan, talaba bu obyekt deb qaralishi mumkin tuzilma.

*Maydon* - bu obyektlarning attributlari yoki xususiyatlarini ifodalovchi tushuncha bo‘lib, sonli yoki son bo‘lmagan qiymatlarni o‘zlashtirishi mumkin.

*Yozuv* - bu bironta obyektga tegishli turli toifadagi maydonlar to‘plamidir.

*Fayl* - bu bir-biriga bog‘liq bo‘lgan yozuvlar to‘plamidir. Masalan, barcha talabalar xaqidagi yozuvlarni o‘z ichiga olishi mumkin,

*Kalit* - bu yozuvdagi maydon bo‘lib, aynan shu yozuvni boshqa yozuvlardan ajratib turishga xizmat qiladi, uning qiymati boshqa yozuvlarda takrorlanmas hisoblanadi. Ba’zida bittadan ko‘p maydonlar qiymatlari elementlararo betakror bo‘lishi mumkin va bunga karrali kalit deyiladi. Ko‘pincha asosiy kalit hisoblanadigan bitta maydon ma’lumoti ishlatiladi va u boshlang‘ich kalit deyiladi, qolganlari esa alternativ kalit deyiladi. Ba’zida esa yozuvlaning yagona qiymatlatli kalit maydonni yo‘qligi sababli kalit sifatida bir nechta maydonlar olinadi va ularga tarkibli kalit deyiladi. Eng yomon holatda, ba’zan shunaqa bo‘lishi mumkinki, bironta maydon kalit bo‘la olmasa, har bit elementga qo‘shimcha, qiymati yagona bo‘lgan kalit maydon kiritiladi.

*Ma’lumotlar toifasi* - qandaydir qiymatlar yig‘indisi bo‘lib, ular ustida ma’lum amallar o‘rinli bo‘ladi. Ma’lumotlar toifalari dasturda oldindan aniqlangan yoki foydalanuvchi tomonidan aniqlangan bo‘lishi mumkin va quyidagi aspektlarni nazarda tutadi.

1. Qiymatlar to‘plami

2. Amallar to‘plami

Misol uchun *int* - butun toifalar va ustida bajariladigan arifmetik amallar (+,-,\*,/).

Ma’lumotlar toifalari 3 turga ajratiladi:

1. *Primitiv toifalar (ma’lumotlarning sodda toifalari)*. Oldindan ma’lum bo‘ladigan, sozlangan toifalar deb xam ataladigan toifalar bo‘lib, turli dasturlash tillarida turlicha bo‘lishi mumkin. Masalan, C++ tilida int (long, short,... ), float(double), char,...

2. *Foydalanuvchi tomonidan aniqlanadigan toifalar*, qachonki mavjud sozlangan toifalar qo‘yilgan masalani yechishga yetarli bo‘lmasa qo‘llaniladi.

3. *Abstrakt toifalar.* Ma’lumotlar toifalarining mantiqiy xususiyatlarini aniqlashda foydali instrument hisoblanadi. “Abstrakt toifa” atamasi bazaviy matematik tushunchaga bog‘liq. Ushbu toifalardagi ma’lumotlar qisman apparat va dasturiy ta’minot yordamida tuzilma sifatida fizik amalga oshirilishi mumkin.

Biz abstrakt toifalarni matematik tushuncha sifatida aniqlaganimizda, muhit va vaqtiy munosabatlarni e’tiborga olmaymiz. Bular amalga oshirish masalalari hisoblanadi.

Ma’lumotlar turli yo‘lar asosida tashkil etilishi mumkin, mantiqiy yoki matematik modelni tashkil etilishi ma’lumotlar tuzilmasi deyiladi. Konkret bir ma’lumotlar tuzilmasini tanlash quyidagilarga bog‘liq:

-Real voqe’likda elementlararo munosabatni yaqqol ifodalay olishi kerak;

-U shunday soda tuzilishi kerakki, zarur bo‘lganda ustida samarali amal bajarish mumkin bo‘lsin.

Ma’lumotlar tuzilmasini o‘rganish quyidagilardan iborat:

- Tuzilmani mantiqiy ifodalash;

- Tuzilmani fiizik amalga oshirish;

-Tuzilmani sifatiy taxlili, ya’ni elementlarni saqlash uchun qancha xotira hajmi sarflanishini aniqlash (xotira sarfi) va qayta ishlashga ketadigan vaqtni (vaqt sarfi) hisoblash nazarda tutiladi.

***Vaqt sarfi.*** Tuzilma ustida amal bajarish algoritmini bajarilish vaqtini hisoblash ko‘zda tutiladi. Buni hisoblashda mashxur katta “O” notatsiya tushunchasi ishlatiladi.

***Xotira sarfi.*** Kirish ma’lumotlarini inobatga olmagan holda, ishlatilayotgan algoritm uchun xotiradan talab qilinadigan qo‘shimcha joy sarfi tushuniladi. Bunda xam katta “O” notatsiyasi qo‘llaniladi.

Vaqt va xotira sarfini hisoblash uchun quyidagi yondashuvlar mavjud:

Katta O notatsiya*. f(x)=O(g(n))* deb belgilanadi, faqat va faqat shunday musbat c va m konstanta mavjud bo‘lib*, f(n)<=c\*g(n)* tengsizlik o‘rinli bo‘lsa, barcha *n, n>=m* holatlarda.

Masalan, ushbu funksiyani *3n+2=O(n)* deb olish mumkin, chunki *3n+2<=4n, n>=2* tengsizlik o‘rinli.

Ushbu funksiyani *6\*2n+n2=O(2n)* deb olish mumkin, chunki *6\*2n+n2<=7\*2n* ifoda o‘rinli, barcha n>=4 larda. *O(1)* deb hisoblash vaqti o‘zgarmas bo‘lgan holatni belgilaymiz. O(n2) ni kvadratik, *O(n3)* ni kubik, *O(2n)* ni eksponensial deb ataladi. Agar algoritmni bajarilish vaqti *O(log n)* bo‘lsa, *O(n)* ga qaraganda tezkor algoritm deb hisoblanadi.

***Omega notatsiya*.** *f(x) = Q(g(n))* deb belgilanadi, faqat va faqat shunday musbat c va m konstanta mavjud bo‘lib, *f(n)<=c\*g(n)* tengsizlik o‘rinli bo‘lsa, barcha n, n>=m holatlarda.

Masalan, *3n+2=Q(n)* deb belgilash mumkin, chunki *3n+2>=3n, n>=1* tengsizlik o‘rinli. *6\*2n+n2=Q(2n)* deb olish mumkin, chunki *6\*2n+n2>=6\*2n* ifoda o‘rinli, barcha *n>=1* larda.

*Teta notatsiya*. *f(x) = 0 (g(n))* deb belgilanadi, faqat va faqat shunday musbat c va m konstanta mavjud bo‘lib, *c\*g(n)<= f(n)<=c2\*g(n)* tengsizlik o‘rinli bo‘lsa, barcha *n, n>=m* holatlarda.

Masalan, 3n+2= 0 (n) deb belgilash mumkin, chunki *3n+2>=3n, n>=1* va *3n+2<=4n* barcha *n>=2* da tengsizlik o‘rinli. *6\*2n+n2= 0 (2n)* deb olish mumkin.

*Algoritmlar samaradorligini hisoblash:*

Algoritmlar samaradorligini hisoblashda kirish ma’lumotini qanday tanlash ko‘rilayotgan algoritmni bajarilishiga yaxshigina ta’sir ko‘rsatadi. Masalan, agar kirish ma’lumotlari allaqachon saralangan bo‘lsa, ba’zi saralash algoritmlari juda yaxshi ishlaydi, ayrimlari ancha past samaradorlik bilan ishlashi mumkin. Agar kirish ma’lumotlari saralanmagan, tartibsiz bo‘lsa, buni aksi bo‘lishi mumkin. Shuni e’tiborga olgan holda, algoritmlar tahlil qilinishi kerak.

***Eng yaxshi holat.*** Bunda kirish ma’lumotlari algoritm tez bajarilishi uchun qulay ko‘rinishda bo‘ladi, ya’ni algoritm kam sonli amallar bilan bajariladi va kam vaqt talab qiladi. Misol uchun, agar tuzimadan qidirayotgan element tuzilmaning birinchi elementi bo‘lib hisoblansa, uni qidirishga eng kam vaqt sarflanadi. Chunki tuzilmaning uzunligidan qat’iy nazar bitta solishtirish yetarli. Algoritmlarni eng yaxshi holatlarini taxlil qilishda odatda, bajarilish vaqti konstanta 1 ga teng bo‘lishi sababli ko‘pincha taxlillarda bu vaziyat ko‘rilmaydi.

***Eng og‘ir holat.*** Bunda kirish ma’lumotlari algoritm bajarilishi uchum eng yomon holatda bo‘ladi va juda sekin bajariladi. Eng og‘ir holat tahlilda muxim hisoblanadi, chunki bu algoritm bajarilishi uchun ketishi mumkin bo‘lgan maksimal vaqtni tasavvur qilishimizga sabab bo‘ladi. Misol uchun, qidirilayotgan element tuzilmaning oxirgi elementi bo‘lsa, uni toppish uchun barcha solishtirishlar amalga oshiriladi.

***O‘rtacha holat.*** Bunda algoritmning o‘rtacha ishlash imkoniyatini beruvchi kirish ma’lumotlari to‘plami olib qaraladi.

Ma’lumotlar tuzilmalari ustida quyidagi amallarni bajarish mumkin:

1. *Ko‘rikdan o‘tkazish (traversing)* - tuzilma elementlariga 1 martadan murojaat qilish amali.

2. *Kiritish* - tuzilmaga yangi element kiritish amali.

3. *O‘chirish* - tuzlmadan bironta elementni o‘chirish amali. Bunda element shunday o‘chirilishi kerakki, qolgan elementlar stabil holatda bo‘lishi kerak, ya’ni ayrim tuzilmalarda nosozlik sezilishi kerak emas.

4. *Qidirish* - tuzilmadan bironta elementni joylashgan o‘rnini aniqlash amali.

5. *Saralash* - elementlarni ma’lum bir tartibda joylashtirish amali.

6. *Birlashtirish (merging)* - ikkita tuzilmani birlashtirish amali.

***Algoritm***. Bironta masalani echish uchun mo‘ljallangan amallarning ma’lum ketma-ketligi hisoblanadi. Algoritmlar quyidagi tamoyillarga asoslanishi kerak:

1. *Kiritish* - bo‘sh qiymat yoki bir nechta qiymatlarni kiritish mumkin bo‘lishi;

2. *Chiqarish* - kamida bitta qiymat chiqarilishi;

3. *Aniqlik* - har bir amal aniq va bitta ma’noga ega bo‘lishi;

4. *Cheklilik* - algoritm chekli sondagi amallardan tashkil topishi;

5. *Samaradorlilik* - har bit amal oddiy va soda bo‘lishi kerak.

Algoritm samaradorligining asosiy ikkita o‘lchami bu vaqt va xotira xajmi hisoblanadi.

***Ma’lumotlar tuzilmasi (MT)*** - information obyektning umumiy xossasi bo‘lib, mazkur xossa bilan biror bir dastur o‘zaro aloqador bo‘ladi. Ushbu umumiy xossa quyidagilar orqali tavsiflanadi:

1) mazkur tuzilmaning mumkin (qabul qilishi mumkin) bo‘lgan qiymatlari to‘plami;

2) mumkin bo‘lgan amallar (operatsiyalar) majmuasi;

3) tashkil etilganlik tasnifi.

**2. Ma’lumotlarni tasvirlash bosqichlari**

Odatda, ma’lumotlarni tasniflash quyidagi ko‘rinishdagi bosqichlarga ajratiladi:

*1) abstrakt (matematik) bosqich;*

*2) mantiqiy bosqich;*

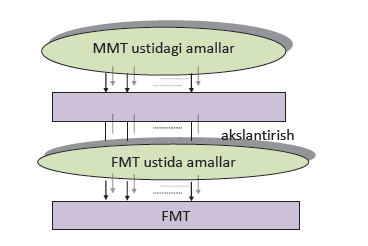
*3) fizik (jismoniy) bosqich.*

Ma’lumki, ixtiyoriy obyekt, xodisa yoki biror bir jarayon tadqiq qilinayotganda uning modeli qurib olinadi. Model turlicha bo‘lishi mumkin, masalan, matematik model, fizik model va boshqa modellar. Obyekt, xodisa yoki biror bir jarayonni matematik model qurildi degani o‘sha qaralayotgan tizimni ma’lum bir matematik qonuniyatlar orqali, ya’ni matematik formulalar orqali ifodalanishidir.

Mantiqiy bosqichda ma’lumotlar tuzilmasini biror bir dasturlash tilida ifodalanishi tushuniladi.

Fizik(jismoniy) bosqichda esa in formats'on obyektni mantiqiy tavsiflanishiga mos ravishda Kompyuter xotirasida akslantirilish tushiniladi. Kompyuter xotirasi chekli bo‘lganligi sababli, xotirani taqsimlash va uni boshqari muammosi yuzaga keladi.

Yuqoridan ko‘rinib turibdiki, mantiqiy bosqich bilan fizik bosqichlar bir biridan farq qiladi. Shu sababli, hisoblash tizimlarida mantiqiy bosqichni fizik bosqichga va aksincha, fizik bosqichni mantiqiy bosqichga akslantirish muamosi vujudga keladi.



*1-rasm. MMT ustida amallar*

Bu yerda MMT - mantiqiy ma’lumotlar tuzilmasi; FMT - fizik ma’lumotlar tuzilmasi;

Abstrakt bosqichda ihtiyoriy tuzilmani <D,R> juftlik korinishda ifodalash mumkin, bu erda D - elementlarning chekli to‘plami bo‘lib, ular, ya’ni elementlar ma’lumotlar turlari yoki ma’lumotlar tuzilmasi bo‘lishi mumkin, R

- esa munosabatlar to‘plami bo‘lib, mazkur munosabatlar hususiyatlari abstrakt bosqichda ma’lumotlar tuzilmalarini turlarini aniqlaydi.

**3. Ma’lumotlar tuzilmasini klassifikatsiya qilish**

Ma’lumotlar tuzilmasini asosiy ko‘rinishlari (turlari):

1) *To‘plam* - munosabat to‘plami bo‘sh R=0 bo‘lgan elementlar majmuasi.

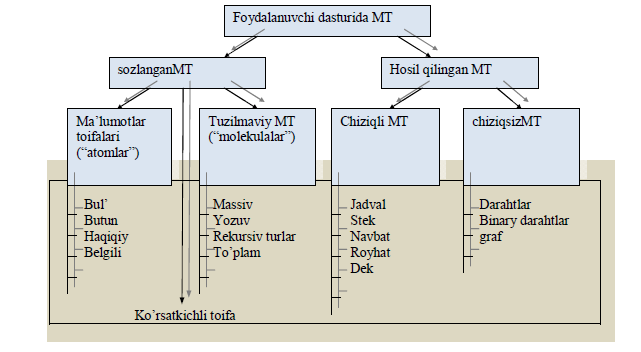
2) *Ketma-ketlik* - shunday abstrakt tuzilmaki, bunda R to‘plam faqatgina bitta chiziqli munosabatdan iborat (ya’ni, birinchi va ohirgi elementdan tashqari har bir element uchun o‘zidan oldin va keyin keladigan element mavjud.

3) *Matritsa* - shunday tuzilmaki, bunda R munosabatlar to‘plami ikkita chiziqli munosabatdan tashkil topgan bo‘ladi.

4) *Daraxt* - bunda R to‘plam iyerarxik tartibdagi bitta munosabatdan tashkil topgan bo‘ladi.

5) *Graf* - bunda R munosabatlar to‘plami faqatgina bitta binar tartibli munosabatdan tashkil topgan bo‘ladi.

6) *Gipergraf* - bu shunday ma’lumotlar tuzilmasiki, bunda R to‘plam ikki yoki undan ortiq turli tartibdagi munosabatlardan tashkil topgan bo‘ladi. Foydalanuvchi dasturida va kompyuter hotirasida MT klassifikatsiya qilish.



*2-rasm. MT klassifikatsiyasi*

MT klassifikatsiya qilishda asosiy belgi bu ma’lumotlar tuzilmasini dastur ishlashi mobaynida o‘zgarishi hisoblanadi. Masalan, agar dastur bajarilishi mobaynida elementlar soni va/yoki ular orasidagi munosabatlar o‘zgarsa, u holda bunday MT dinamik ma’lumotlar tuzilmasi, aks holda statik ma’lumotlar tuzilmasi deyiladi.

Ma’lumotlar tuzilmasiga misollar:

Ma’lumki, matematikada o‘zgaruvchilarni, ularning ba’zi bir kerakli tavsiflariga mos ravishda klassifikatsiya qilish qabul qilingan. O‘zgaruvchilarga misol sifatida quyidagilarni keltirib o‘tish mumkin: haqiqiy o‘zgaruvchilar, kompleks o‘zgaruvchilar, mantiqiy o‘zgaruvchilar, bundan tashqari ba’zi bir qiymatlarni qabul qiluvchi o‘zgaruvchilar va boshqalar. Ma’lumotlarni qayta ishlashda ularni klassifikatsiya qilish ham katta ahamiyatga ega. Bu yerda ham klassifikatsiya qilinayotganda har bir konstanta, o‘zgaruvchi, ifoda yoki funksiya biror bir toifarga tegishli bo‘ladi degan tamoyilga asoslanadi.

1. **Ma’lumotlarni abstrakt toifalari.**

Umuman olganda toifalar o‘zgaruvchi yoki ifoda qabul qilishi mumkin bo‘lgan qiymatlar to‘plami orqali tavsiflanadi.

Ko‘plab dasturlash tillarida ma’lumotlar standart va foydalanuvchi tomonidan beriladigan toifalar ga ajratiladi. Ma’lumotlarni standart toifalariga quyidagi 5 ta tur o‘zgaruvchilari kiradi:

*a) butun (INT);*

*b) haqiqiy (FLOAT) ;*

*c) mantiqiy (BOOL);*

*d) belgili (simvol) (CHAR);*

*e) ko‘rsatkichli (\*).*

Foydalanuvchi tomonidan aniqlanadigan toifalar esa:

*a) sanaladigan;*

*b) diapazonli (oraliqli).*

Ma’lumotlarning ixtiyoriy toifasi qiymatlar sohasi va ular ustida bajarilishi mumkin bo‘lgan amallar orqali tavsiflanadi.

***Butun toifa – INT.*** Mazkur toifa butun sonlar to‘plamini qandaydir qism to‘plami bo‘lib, uning o‘lchami mashina, ya’ni EHM konfiguratsiyasiga bog‘liq ravishda o‘zgarib turadi. Agar butun sonni mashinada tasvirlash uchun p ta razryaddan foydalanilsa (bunda qo‘shimcha koddan foydalanilganda), u holda x butun sonning qiymat qabul qilish oralig‘i quyidagicha bo‘lishi zarur, ya’ni quyidagi shartni qanoatlantirishi lozim:

*-2 n-1<= x< 2 n-1.*

Butun toifadagi ma’lumotlar ustida bajariladigan barcha amallar to‘g‘ri amalga oshiriladi deb hisoblanib, ushbu amallar arifmetikada qabul qilgan qoidalariga bo‘ysunadi. Agar ushbu toifada amallar bajarilganda natija ruxsat etilgan oraliqdan chiqib ketsa, u holda hisoblash to‘xtatiladi. Bunday hol to‘lib ketish deb ataladi.

Mazkur toifaga kiruvchi sonlar ikkitaga bo‘linadi: ishorali va ishorasiz. Ularning har bir uchun mos ravishda qiymat qabul qilish oralig‘i mavjud:

a) ishorasiz sonlar uchun *(0..2n-1);*

b) ishoralilar uchun *(-2N-1.. 2N-1-1).*



*3-rasm. Butun toifa*

Sonlar mashinada qayta ishlanayotganda ularning ishorali ko‘rinishidan foydalaniladi. Agar mashina so‘zi yozuv, komandarani qayta ishlash va ko‘rsatkichlar uchun foydalanilayotgan bo‘lsa, u holda sonning ishorasiz ko‘rinishidan foydalaniladi.

Butun sonlar ustida - qo‘shish, ayrish, ko‘paytirish, butunsonli bo‘lish (qoldiqni tashlab yuborish orqali), berilgan modul bo‘yicha hisoblash (bo‘lishda qolgan qoldiqni hisoblash), berilgan sonlar to‘plamining eng katta va eng kichik elementini aniqlash, butun darajaga oshirish, sonning qiymatiga qarab o‘zidan oldingi yoki keyingi sonni aniqlash. Bu operatsiyalarning natijalari ham butun sonlar bo‘ladi.

Butun sonlar ustida ==,!=, <, <=, >, >= operatorlar bilan taqqoslash amallarni ham bajarish mumkin. Ammo bu operatsiyalarning natijalari INT toifasiga kirmaydi, ular BOOL toifasiga kiradi.

***Haqiqiy toifa – FLOAT.*** Haqiqiy toifaga kasr qismlari bor chekli sonlar to‘plami kiradi. To‘plamni chekli bo‘lish sharti kompyuterda sonlarni ifodalash chegaralanganligi bilan bog‘liq. Haqiqiy sonlar ustida quyidagi amallarni bajarish mumkin: qo‘shish, ayrish, bo‘lish, ko‘paytirish, trigonometrik funksiyalarini hisoblash, darajaga oshirish, kvadrat ildiz chiqarish, logarifmlash, minimum va maksimum elementlarni topish va boshqalar. Bularning natijalari ham haqiqiy toifaga kiradi. Bu yerda ham binar amallarga nisbatan masalaning yechimlari mantiqiy toifaga tegishli bo‘ladi.

Kompyuter xotirasida haqiqiy sonlar asosan qo‘zg‘aluvchan nuqta formatida saqlanadi. Bu formatda x haqiqiy son quyidagi ko‘rinishda ifodalanadi:

x = +/- M \* q(+/-P) - soning yarimlogarifmik shakldagi ifodalanishi quyidagi chizmada keltirilgan.

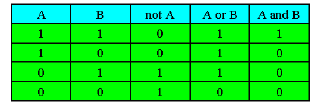
937,56 = 93756 \* 10-2 = 0,93756 \* 103



*4-rasm. Haqiqiy toifa*

***Mantiqiy toifa – BOOL.*** Operatsiyalarni bajarish mumkin: kon’yunksiya (va), diz’yunksiya (yoki) i inkor (yo‘q), hamda qiyinroq bo‘lgan ekvivalentlik, implikatsiya, chiqarib tashlash, yoki va boshqa operatsiyalar. Yuqorida keltirilgan ixtiyoriy operatsiyaning natijasi - mantiqiy qiymatga ega bo‘ladi. Mantiqiy qiymatni xotirada saqlash uchun bitta bit yetarli.

Asosiy mantiqiy funksiyalarning chinlik jadvali



*5-rasm. Mantiqiy amallar jadvali*

***Belgili toifa – String.*** Belgili toifaga belgilarning chekli to‘plami yoki liter, ularga lotin alifbosidagi harflar va unda yo‘q kirill harflar, o‘nlik raqamlar, matematik va maxsus belgilar kiradi. Belgili ma’lumotlar hisoblash texnikasi bilan inson o‘rtasidagi aloqani o‘rnatishda katta ahamiyatga ega. Ko‘pincha, dasturlashning har bir tizimida belgilar to‘plami fiksirlangan bo‘lib, ular turli tizimlarda turli hil bo‘lishi mumkin. Bundan tashqari ular tartiblangan bo‘lib, har bir uning elementiga aniq bir sonli kod mos qo‘yilib, u to‘plamdagi tartib raqamini aniqlaydi. Belgini sonli kodiga o‘tib, relyatsion operatorlardan foydalanib, simvollarni taqqoslash mumkin.Bunday taqqoslashlarning natijalari BOOL toifasiga kiradi.

C++ tilida belgili toifadan tashqari belgilar massividan tashkil topgan satrli toifalar bilan xam ishlash mumkin, ya’ni char []. Shu o‘rinda aytib o‘tish kerakki, satrlar bilan ishlashda belgilar massividan tashqari satrlar bilan ishlashga mo‘ljallangan maxsus kutubxona mavjud bo‘lib, String deb nomlanadi. Satr (qator, String) - bu qandaydir belgilar ketma-ketligi. Satr bitta,

bo‘sh yoki bir nechta belgilar birlashmasidan iborat bo‘lishi mumkin. C++ tilida satr 0 dan to 255 tagacha uzunlikka ega bo‘lishi mumkin. Agar o‘zgaruvchi satr toifasiga tegishli bo‘lsa, u holda o‘zgaruvchi toifasi yozilayotganda 2 xil ko‘rinishda char [] yoki ***String*** deb aniqlanadi.

Belgili toifadagi amallar:

*a) O‘zlashtirish;*

*b) Taqqoslash;*

***Ko‘rsatkichli toifa (Pointer).*** Ko‘rsatkichli toifa ma’lumotlarni ko‘rsatkichlari yoki manzillari (adres) to‘plamini namoyon qiladi, ya’ni ko‘rsatkichlar ma’lumotlarni emas balki bu ma’lumotlar joylashgan xotiradagi manzilni o‘z ichiga oladi. Ko‘rsatkichlar xotirada bori yo‘g‘i 4 bayt joyni egallab, u ko‘rsatayotgan ma’lumotlar ancha katta joyni egallagan bo‘lishi mumkin. Pointer toifasi ma’lumoti ixtiyoriy boshqa biror ma’lumot yoki ma’lumotlar guruhiga yo‘naltirilgan bo‘ladi. Ko‘rsatkichga mumkin bo‘lgan u yoki bu qiymatni o‘zlashtirib, ushbu ko‘rsatkich orqali kerakli ma’lumotga murojatni amalga oshirish mumkin. Pointer toifasidagi ma’lumotlarni qiymatlar to‘plamida bitta maxsus qiymat bo‘lib, uni o‘zlashtirish hech qayerga yo‘naltirilmaganligini ko‘rsatadi, ya’ni nol yoki bo‘sh ko‘rsatkich hisoblanadi.

Masalan, C++ tilida bunday qiymat sifatida NULL dan foydalaniladi. Ko‘rsatkichlar ustida amallar quyidagicha bo‘lishi mumkin: biror bir ko‘rsatkichga boshqa ko‘rsatkich qiymatini o‘zlashtirish mumkin yoki boshqa ma’lumot egallab turgan xotira sohasi adresini o‘zlashtirish mumkin. Ko‘rsatkichlar o‘zaro bog‘langan ma’lumotlar tuzilmasini yaratishda va qayta ishlashda katta ahamiyatga ega. Xotirada ko‘rsatkichlarni ifodalash uchun uchun asosan dasturlash tizimiga mos ravishda manzilni maksimal uzunligicha joy ajratiladi. Ko‘rsatkichlarni qiymati nomanfiy butun sonlar sifatida sohada bitlarni ketma-ketligi ko‘rinishida saqlanadi.C++ tilida ko‘rsatkichli o‘zgaruvchilarni e’lon qilish uchun ularning toifasini aniqlash kerak. Buning uchun ko‘rsatkich xotirada qanaqa toifadagi ma’lumotlarni ko‘rsatayotgan bo‘lsa, ko‘rsatkichli o‘zgaruvchiga xam xuddi shunday toifa beriladi.

*int a=9;*

*int \*p=&a;*

*float f=4.6;*

*float \*d=&f;*

*FILE \*f=fopen(“talaba.txt”,’r’);*

Foydalanuvchi tomonidan aniqlanadigan toifalar Sanaladigan toifalar

Qiymatlarning o‘zgaruvchan toifalari standartlardan farqliroq yangi toifalarni yaratishga imkon beradi. Bu guruhga sanaladigan va chegaralangan toifalar kiradi.

Qiymatlarning sanaladigan toifalarning bunday atalishiga sabab, ular qat’iy aniqlangan tartibda sanaladigan ko‘rinishda beriladi va xamma qiymatlarning soni qat’iy chegaralangan xamda ko‘rilayotgan toifadagi qiymatlarni qabul qilishi mumkin. Sanaladigan toifa yechilayotgan masalaga qarab foydalanuvchi tomonidan berilishi mumkin.

Sanaladigan toifa konstantalar ro‘yxatidan tashkil topadi. Bu toifadagi o‘zgaruvchilar ro‘yxatidagi ixtiyoriy qiymatni qabul qilishi mumkin. Sanaladigan toifaning umumiy yozilish shakli quyidagicha: enum toifaning nomi (konstantalar ro‘yxati}; toifaning nomi o‘zgaruvchi nomi;

Bu yerda konstanta tushunchasi foydalanuvchi tomonidan berilagan maxsus konstanta ko‘rinishi tushuniladi. Konstantalar ro‘yxati bir-biridan vergul bilan ajratiladi va ular oddiy qavslar ichiga olinadi.

***Masalan***:

*enumRanglar(oq, qora, qizil, yashil};*

*Ranglar rang;*

Bu yerda Ranglar - sanaladigan toifaning nomi; oq, qora, qizil, yashil- konstantalar. Rang - o‘zgaruvchi nomi bo‘lib u yuqoridagi konstantalardan ixtiyoriysini qabul qilishi mumkin.

Har bir konstanta tartib raqamiga ega bo‘lib, hisobdan boshlanadi, ya’ni oq=0, qora=1, qizil=2, yashil=3 raqamlariga ega. Konstantalar tartiblangani uchun ularga solishtirish amallari <, <=,==,!=, >=, > shuningdek standart funksiyalarni qo‘llash mumkin.

***Strukturalar.*** Strukturalar turli toifadagi maydonlardan tashkil topgan yozuv hisoblanadi. Strukturalarni e’lon qilish uchun ***struct*** kalit so‘zi ishlatiladi. Undan keyin toifaga nom beriladi va {} qavs ichida maydonlar toifalari va nomlari e’lon qilinadi.

*struct G(*

*charch;*

*} talaba, talabalar[10];*

Ushbu toifadagi o‘zgaruvchiyoki massiv elementi maydonlariga murojaat:

*• Jadval\_elementi[indeks]. maydon\_nomi=qiymat;*

*• Ya’ni, talabalar[i].ch=’a ’;*

**Nazorat savollari**

1. Ma’lumotlar tuzilmasi deganda nimani tushunasiz?

2. Ma’lumotlarni tasvirlash bosqichlarini keltirib o‘ting.

3. Ma’lumotlar tuzilmasiklassifikatsiyasi vafoydalanuvchi dasturidagi klassifikatsiyasi qanday?

4. Ma’lumotlar tuzilmasini operativ va tashqi xotiradagi klassifikatsiyasi.

5. Qanday ma’lumotlar dinamik yoki statik turdagi ma’lumotlar tuzilmasi deyiladi?

6. Ma’lumotlarning qanday toifalarini bilasiz?

7. Butun toifadagi ma’lumotlar ustida qanday amallarni bajarish mumkin?

8. Ma’lumotlarning bul toifasida qanday amallar mavjud?

9. CHAR toifasining tuzilmasi qanday?Belgili toifadan qanday amallarni bajarish mumkin?

10. Ko‘rsatkichli toifa ma’lumoti yordamida nimani hisoblash mumkin?

11. Ma’lumotlarning sanaladigan toifasi degani nima?

12. Struktura toifasi qanday beriladi?