



MUHAMMAD AL-XORAZMIY NOMIDAGI
TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI



DASTURLASH 1

SWD1316

```
def dotwrite(ast):  
    n = getNodeName()  
    n = n.strip() + ".sym_name.get(int(ast[0]), ast[0])"  
    print("%s [label=\"%s\" % (nodename, label),  
    nance(ast[1], str):  
    if ast[1].strip():
```

01

MAVZU

ALGORITMLASH VA
DASTURLASHNING ASOSIY
TUSHUNCHALARI.

```
children = []  
for n, child in enumerate(ast[1:]):  
    children.append(dotwrite(child))  
print '%s -> {' % n  
for name in children:  
    print '%s' % name,
```



ISHNIYAZOV ODIL
OLIMOVICH



Informatika asoslari kafedrası
katta o'qituvchisi



Ma'ruza rejasi



- 1 Tilning bazaviy tushunchalari
- 2 Identifikatorlar, ularga qiymat o'zlashtirish usullari va operatorlari
- 3 Inkrement va dekrement
- 4 Ma'lumotlar toifasini o'zgartirish
- 5 Matematik funksiyalar va kutubxonalar

Tilning bazaviy tushunchalari



Tilning bazaviy tushunchalari



void = «bo'sh»
Kompilyator
turlariga qarab
int ga o'zgarishi
mumkin

Dastur tanasi
(yordamchi
funksiya ham
chaqiriladi)

```
void  
main ()  
{  
  
  
}
```

Dasturda kamida
1 ta funksiya
main() bo'lishi
shart

Blok – dasturning
bir qismi
(boshlanishi)

Blok – dasturning
bir qismi
(yakunlash)

Tilning bazaviy tushunchalari



Mumkin bo'lgan belgilar:

Katta va kichik lotin alifbosi (A-Z, a-z)

Ta'kidlovchi belgi _

Raqamlar 0,1,2,3,4,5,6,7,8,9

Maxsus simvollar: “, { } | [] () + - / % \ ; ‘ . : ? < = > _ ! & *

~ ^



Nomni raqamlar bilan boshlash mumkin emas!

Mumkin bo'lmagan belgilar:

Rus alifbosi.

Probel.

Qavs, turli belgilar, +, -, ! va x.k.

Identifikatorlar, ularga qiymat o'zlashtirish usullari va operatortlari



Identifikatorlar lotin harflari, ostki chiziq belgisi va sonlar ketma-ketligidan iborat bo'ladi. Identifikator lotin harfidan yoki ostki chiziq belgisidan boshlanishi lozim.

Misol uchun:

A1, _MAX, adress_01, RIM, rim

Katta va kichik harflar farqlanadi, shuning uchun oxirgi ikki identifikator bir-biridan farq qiladi.

Borland kompilyatorlaridan foydalanilganda nomning birinchi 32 harfi, ba'zi kompilyatorlarda 8 ta harfni inobatga oladi.

Bu holda **NUMBER_OF_TEST** va **NUMBER_OF_ROOM** identifikatorlari bir biridan farq qilmaydi.

Xizmatchi so'zlar



Tiplar nomlari:

char, short, int, unsigned, long, float, double.

Operatorlar nomlari:

if, else, switch, case, while, do, for, default, break, continue, goto.

Xotira turlari:

auto, register, static, extern.

Tiplar bilan ishlash:

typedef, sizeof.

Struktura: *struct, union.*

Chiqish: *return, entry.*

O'zgaruvchi nima?



O'zgaruvchi - ma'lum nomga ega va o'zida qiymatlarni saqlaydigan kompter xotirasidagi yacheyka.

Dastur bajarilish jarayonida o'zgaruvchilar qiymati o'zgarishi mumkin.

Yacheykaga yangi qiymat yozilganda eskisi o'chib ketadi.

O'zgaruvchi tiplari:

int – [-32768...32767] oraliqdagi butun sonlar (2 baytgacha)

float – xaqiqiy son, floating point (4 baytgacha)

char – belgi, character (1 baytgacha)

O'zgaruvchilarning tiplari



1. **char** – bitta simvol;
2. **long char** – uzun simvol;
3. **int** – butun son;
4. **short** yoki **short int** – qisqa butun son;
5. **long** yoki **long int** – uzun butun son;
6. **float** haqiqiy son;
7. **long float** yoki **double** – ikkilangan haqiqiy son;
8. **long double** – uzun ikkilangan haqiqiy son.

O'zgaruvchilarni e'lon qilish va qiymat berish



O'zgaruvchilarni dasturning ixtiyoriy qismida qayta ta'riflash mumkin. Misol uchun:

int a, b1, ac; yoki

int a; int b1; int ac;

O'zgaruvchilar ta'riflanganda ularning qiymatlari aniqlanmagan bo'ladi. Lekin o'zgaruvchilarni ta'riflashda initsializatsiya ya'ni boshlang'ich qiymatlarini ko'rsatishi mumkin.

Misol uchun:

int i = 0;

char c = 'k';

Konstanta(o'zgarmas)lar



Konstanta - o'zgartirish mumkin bo'lmagan qiymat. C tilida besh turdagi konstantalar ishlatilishi mumkin:

simvollar, butun sonlar, haqiqiy sonlar, sanovchi konstantalar va nul ko'rsatkich.

Belgili o'zgarmaslar - odatda bir bayt joyni egallaydi va bu 256 xil belgini saqlash uchun yetarlidir. **Char** tipi qiymatlarini 0..255 sonlar to'plamiga yoki ASCII belgilar to'plamiga interpretatsiya qilish mumkin.

ASCII belgilari



ASCII belgilari – kompyuterdagi standart belgilar to‘plami.

ASCII - bu American Standard Code for Information Interchange (Amerikaning axborot almashinishi uchun standart kodi) degan ma'noni anglatadi. **Misol uchun:**

`'x', '*', '\012', '\0', '\n'` - bitta simvolli konstanta;

`'dd', '\n\t', '\x07\x07'` ikki simvolli konstantalar.

C++ kompilyatorida tekstlarni formatlovchi bir nechta maxsus belgilardan foydalaniladi. Maxsus belgilar axborotlarni ekranga, faylga va boshqa chiqarish qurilmalariga chiqarishda formatlash uchun qo'llaniladi.

Nomlangan konstantalar



C++ tilida o'zgaruvchilardan tashqari nomlangan konstantalar kiritilishi mumkin. Bu konstantalar qiymatlarini dasturda o'zgartirish mumkin emas. Konstantalar nomlari dasturchi tomonidan kiritilgan va xizmatchi so'zlardan farqli bo'lgan identifikatorlar bo'lishi mumkin. Odatda nom sifatida katta lotin harflari va ostiga chizish belgilari kombinatsiyasidan iborat identifikatorlar ishlatiladi. Nomlangan konstantalar quyidagi shaklda kiritiladi:

const tip konstanta_nomi = konstanta_qiymati.

Nomlangan konstantalar



Misol uchun:

```
const double EULER = 2.718282;
```

```
const long M = 99999999;
```

```
const R = 765;
```

Oxirgi misolda konstanta tipi ko'rsatilmagan, bu konstanta **int** tipiga tegishli deb hisoblanadi.

Inkrement va dekrement



Inkrement `++` unar amali qiymatni **1** ga oshirishni ko'rsatadi. Amalni prefiks, ya'ni `++i` ko'rinishda ishlatish oldin o'zgaruvchi qiymatini oshirib, so'ngra foydalanish lozimligini, postfiks esa `i++` ko'rinishda ishlatish oldin o'zgaruvchi qiymati foydalanib, so'ngra oshirish kerakligini ko'rsatadi. Misol uchun, `i` ning qiymati **2** ga teng bo'lsin, u holda `3+(++i)` ifoda qiymati **6** ga, `3+i++` ifoda qiymati **5** ga teng bo'ladi. Ikkala holda ham `i` ning qiymati **3** ga teng bo'ladi.

Inkrement va dekrement



Dekrement -- unar amali qiymati 1 ga kamaytirishni ko'rsatadi. Bu amal ham prefiks va postfiks ko'rinishda ishlatilishi mumkin. Bu ikki amalni faqat o'zgaruvchilarga qo'llash mumkin.

Qiymat berish amali



Bundan tashqari C++ tilida murakkab qiymat berish amali mavjud bo‘lib, umumiy ko‘rinishi quyidagichadir:

O‘zgaruvchi_nomi amal = ifoda;

Bu yerda amal quyidagi amallardan biri $*, /, \%, +, -, \&, ^, |, <<, >>.$

Misol uchun:

$x+ = 4$ ifoda $x = x+4$ ifodaga ekvivalentdir;

$x* = a$ ifoda $x = x*a$ ifodaga ekvivalentdir;

$x/ = a+b$ ifoda $x = x/(a+b)$ ifodaga ekvivalentdir;

Shartli amal



Shartli amal ternar amal deyiladi va uchta operanddan iborat boʻladi:

<1-ifoda>?<2-ifoda>:<3-ifoda>

Shartli amal bajarilganda avval 1- ifoda hisoblanadi. Agar 1-ifoda qiymati 0 dan farqli boʻlsa 2- ifoda hisoblanadi va qiymati natija sifatida qabul qilinadi, aks holda 3-ifoda hisoblanadi va qiymati natija sifatida qabul qilinadi.

Misol uchun modulni hisoblash: **$x < 0 ? -x : x$** yoki ikkita sondan kichigini topish **$a < b ? a : b$** .

Munosabat va mantiqiy ifodalar



- **Munosabat amallari.**

$==$ - teng, $!=$ - teng emas

$<$ - kichik, $<=$ - kichik yoki teng

$>$ - katta, $>=$ - katta yoki teng

- **Mantiqiy amallar**

$!$ - inkor

$\&\&$ - kon'yunksiya (mantiqiy ko'paytirish)

$||$ - diz'yunksiya (mantiqiy qo'shish)

Mantiqiy amallar jadvali



x	y	$\neg x$	$x \vee y$	$x \wedge y$
0	0	1	0	0
0	1	1	1	0
1	0	0	1	0
1	1	0	1	1

Taqqoslash amallari



Amallar	Qo'llanilishi	Mazmuni (o'qilishi)
$<$	$a < b$	“a kichik b”
$\leq$	$a \leq b$	“a kichik yoki teng b”
$>$	$a > b$	“a katta b”
$\geq$	$a \geq b$	“B kata yoki teng b”
$==$	$a == b$	“a teng b”
$!=$	$a != b$	“a teng emas b”

Kutubxonalar

- 1. **#include <stdio.h>** - C tilida kiritish/chiqarish.
- 2. **#include <iostream.h>** - C++ tilida kiritish / chiqarish
- 3. **#include <math.h>** - standart funksiyalar uchun
- 4. **#include <conio.h>** - kiritish/chiqarishni nazorat qilish
- 5. **#include <string.h>** - qator tipli o'zgaruvchilar bilan ishlash
- 6. **#include <stdlib.h>** - standart kutubxona fayllarini chaqirish
- 7. **# include <time.h>** - kompyuterning soatidan foydalanish
- 8. **#include <graphics.h>** - grafik imkoniyatlaridan foydalanish

Matematik funksiyalar



Funksiya	Yozilishi	Funksiya	Yozilishi
$\sin x$	$\sin(x)$	$\sqrt{X}$	$\text{sqrt}(x); \text{pow}(x, 1./2)$
$\cos x$	$\cos(x)$	$ X $	$\text{abs}(x); \text{fabs}(x)$
$\text{tg } x$	$\tan(x)$	$\arctan x$	$\text{atan}(x)$
$\ln x$	$\log(x)$	$\sqrt[3]{X^2}$	$\text{pow}(x, 2./3)$
$\lg x$	$\log_{10}(x)$	$\log_2 x$	$\log(x)/\log(2); \log_2(x)$
x^a	$\text{pow}(x, a)$	e^x	$\text{exp}(x)$
$\arcsin x$	$\text{asin}(x)$	$\arccos x$	$\text{acos}(x)$

Qo'shimcha standart funksiyalar



ceil(x) - x ni o'zidan katta yoki teng bo'lgan butun songacha yaxlitlash.

Masalan: $\text{ceil}(12.6) = 13$; $\text{ceil}(-2.4) = -2$;

floor(x) - x ni o'zidan kichik yoki teng bo'lgan butun songacha yaxlitlash.

Masalan: $\text{floor}(12.1) = 12$; $\text{floor}(4.8) = 4$; $\text{floor}(-12.1) = -13$; $\text{floor}(15.9) = 15$;

fmod(x,y) - x/y ning qoldig'ini kasr son ko'rinishida beradi.

Masalan: $\text{fmod}(7.3, 1.7) = 0.5$

XULOSA



- Hayotda masalalarni yechishda uning algoritmini tuzishdan boshlash maqsadga muvofiq bo'ladi. Masala algoritmini tuzgan shaxs istalgan dasturlashda masalani yechish mumkin bo'ladi, albatta undan shu dasturlash tili sintaksisini bilish kifoya qiladi.
- Masalani yechish algoritmini tuzish, so'zlar, matematik formulalar, grafiklar va jadvallar asosida amalga oshirish mumkin.
- Algoritmnlarni chiziqli, tarmoqlanuvchi va takrorlanuvchi ko'rinishda ifodalash mumkin.



MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI



```
def dotwrite(ast):  
    nodename = getNodeName()  
    label=symbol.sym_name.get(int(ast[0]),ast[0])  
    print '      %s [label="%s' % (nodename, label),  
    isinstance(ast[1], str):  
        if ast[1].strip():  
            p  
        else:  
            p  
        e:  
        print  
    children = []  
    for n, child in enumerate(ast[1:]):  
        children.append(dotwrite(child))  
    print '      %s -> {' % nodename  
    for name in children:  
        print '%s' % name,
```

E'TIBORINGIZ UCHUN RAXMAT!



ISHNIYAZOV ODIL
OLIMOVICH



Informatika asoslari kafedrası
katta o'qituvchisi