

Дәріс - 3.

Тақырыбы: C++Builder бағдарламалау жүйесінің құрамы. Құралдар тақтасы және жүйенің негізгі мәзірі

Жоспар:

1. Тұрақтылар
2. Операциялар
3. Көрсеткіш

Түрлі айнымалылардың, тұрақтылар мен операция символдарының қандай да болмасын бір жинақ түрінде жазылуы өрнек деп аталады. Си тілінде өрнек соңына нүктелі үтір (;) символы қойылып жазылады. Осы түрде жазылған өрнек не функция оператор делінеді. Мысалы, операторлар:

$$z = (3 * x + y + 5); y = \sin(x);$$

Сипаттаманың соңына да нүктелі үтір қойылып жазылады, сондықтан ол да оператор. Мысалы: float y, z; сипаттамасы — оператор. Бір символдық оператор мәні бір не бірнеше қатарларда жазылуы мүмкін, тек онда пайдаланатын символдар саны байтпен көрсетілген аралықтан аспаса болғаны.

Функция C++ тілінде түрлі ұғымды білдіреді. Мысалы:

- 1) блок басында сипатталатын *айнымалы*;
- 2) программа ішінде орындалатын *блок*;
- 3) арнайы ат беріліп, *кітапханада* мәнін есептейтін ішкі программа бойынша жазылып *сақталған* функция, (мысалы, sin(), sqrt(), т.б.).

Мысалдар:

int radius (x);	тип, функция аты және аргумент
{ Операторлар }	құрама оператор – функция
float x, y;	тип, айнымалылар аттары. Сипаттамаға сәйкес x, y айнымалыларының мәндері үшін алты мәндік ондық цифрлар алынады.
y=sin (6.5)+3;	оператор; sin() функциясының мәні үшін арнайы стандартты кітапхана шақырылып, мәні автоматты түрде есептеледі.

(Си тілінде тригонометриялық функциялардың мәндері радиан арқылы өлшенеді).

Мұндағы ескеретін жайт: соңғы өрнек — оператор. Оның соңына қойылған қойылатын нүктені үтір алып тасталған кезде қалған бөлігі ($y = \sin 6.5 + 3$) — функция, ал sin () — осы *функция ішіндегі стандартты функция*. sin (6.5) — өрнегі функцияны шақыру деп аталады.

Тұрақтылар. Түсініктеме

Программаға енгізілген өзгертуге болмайтын шамалардың тұрақты делінетіні белгілі. Тұрақтылар түрлі типті болуы мүмкін.

Мысалы:

Тұрақты	тип
10, 25, - 542	Int
'a', 'b', '15'	char
15.23, 4.2E+5	float

Бүтін тұрақтылар

Программада ондық, сегіздік, он алтылық санау жүйелерінде берілген сандарды пайдалануға болады. *Нөл (0) цифрынан басталған* цифрлар тізбегі (0, 1, 2 ,..., 7) сегіздік жүйеде, ал *нөл емес цифрдан басталған* тізбек ондық жүйеде берілген сан деп есептеледі. Цифрлар тізбегі 0x не 0X - тен бастап жазылған болса, ол он алтылық санау жүйесінде жазылған деп есептеледі. Бұл жүйедегі 10 ... 15 аралығының цифрлары әдеттегідей A, B, ..., F не a, b,..., f символдарымен белгіленеді. Мысалдар:

$$014_8 = 12,0$$

$$14 = 14_{10}$$

$$0x14_{16} = 20_{10}$$

$$0X14_{16} = 20_{10}$$

$$0x188,6 = (1 * 16^2 + 11 * 16 + 8)_{10} = 440_{10}$$

Нақты санның бүтін және бөлшек бөлімдері нүкте арқылы бөлініп жазылады не e(E) экспонентасы арқылы белгіленеді.

Мысалдар:

$$513. = 513$$

$$5.45 = 5,45$$

$$3.6E+4 = 36\ 000$$

$$3.6E-4 = 0,00036$$

Бұлардың алғашқысы бүтін сан емес, нақты сан ретінде қабылданады, себебі соңына нүкте қойылған.

Символдық тұрақтылар

Символдық тұрақты дәйекшелер ішінде жазылады. Мысалы: 'd', 'f', '15'.

Олардың типі - char. Символдық тұрақтыларды **int** типі арқылы сипаттауға да болады, себебі: компилятор олар үшін жадқа символдардың ASCII кодын орналастырады.

C++ тілінде әдеттегідей жолдық тұрақты жоқ, ол символдар тізбегінен тұратын символдық массив ретінде қарастырылады. *Кейде ол жол түрінде де* пайдаланылуы мүмкін. Бұл кезде ол тырнақшалар ішінде жазылады. Жол жеке символдардан тұратын етіп берілсе, символдар ретімен дәйекшелерге алынып жазылады.

Мысалы:

"Bas", "1975 жылы туылған" /*жолдық тұрақтылар */

'B', 'a', 's' /*символдықтұрақтылар*/

Жолды пернетақтадан теру аяқталған кезде Enter пернесін басу керек. Компилятор жолдың аяқталу белгісі болатып әр жолдың соңына '\0' символын орналастырып кетеді.

Программада жолдық тұрақты char ms [60] сияқты түрде сипатталады. Мұндағы char - массив элементінің типі, ms - массив аты, 60 - арнайы қалдырылған массив өлшемі. Оның орнына қажетті басқа сан алуға да болады, тек жол түзетін символдар санынан кемінде 1-ге артық болуы керек. Ол жолдың соңына енгізіліп қойылатын нөл (0) цифрын жазу үшін қажет. Мысалы, "Bas" жолдық тұрақтысының жадта орналасу түрі:

B	a	s	\0
---	---	---	----

Символдық айнымалылармен жұмыс істеу тәсілдері 2.8 - тақырыпқа енгізілген.

Түсініктеме

Түсініп оқуға жеңіл болуы үшін күрделі программалардың қажетті жерлеріне түсініктеме мәтіндер енгізіліп қойылады. Турбо C++ тілінде түсініктеме /* */ символдарының арасына жазылады. Түсініктеме бірнеше қатарлық болуы да мүмкін, ол үшін түсініктеме енгізетін бос орындар болса болғаны.

Мысалдар:

1) /* квадрат теңдеу */

2) /* программаға енгізілген

/* көпжолдық түсініктеменің

/* жазылу түрі */

Операциялар. Көрсеткіш.

C++ тілінде пайдаланылатын негізгі операциялар: арифметикалық, қатыс, логикалық, меншіктеу, биттік және шарт.

Арифметикалық операциялар

Операция	Айнымалы типтері
Қосу (+)	int, char, float
Азайту (-)	int, char, float
Көбейту (*)	int, char, float
Бөлу (/)	int, float
Қалдық (%)	int

Егер операндтар бір типті болса, алдыңғы үш операция басқа алгоритмдік тілдердегі операциялар сияқты пайдаланылады.

Бүтін типті айнымалылардың мәндерін **бөлу** (/) операциясының нәтижесі де бүтін типті (**int**), бірақ ерекшелігі - бұл операция нәтижесінде бөліндінің тек бүтін бөлігі шығарылады да, қалдық алынып тасталады. Мысалы, $17/3=5$; $-17/3= - 5$. Сондықтан программада кездескен бөлшектермен амал істеу кезінде оларды **float** типі арқылы сипаттау керек.

Мысалы,

```
1) float k,y,s; for(k=1; k<=3; k++) y=1/k; s=s+y;
}
```

```
2) float s; s=1.+1./2 +1./3; үзінділерінің нәтижелері бірдей.
```

Қалдық (%) **операциясы** да тек бүтін типті сандарды бөлуде қолданылады: $17\% 3 = 2$; $-17\% 3 = -2$. Мұндай ерекшеліктер төмендегі программада көрсетілген.

```
#include<stdio.h>
#include<conio.h>
main() (2)
{ int x, y, z1, z2; clrscr ();
printf("сандар:x,y=?");
scanf("%d%d",&x,&y);
z1=x % y; z2=x/ y;
printf("қалдық=%d\n", z1);
printf("бөлінді=%d", z2);
getch (); return 0;
}
```

Қатыс операциясы салыстыру үшін шарт түрінде жазылатын операция. C++ тілінде алты қатыс операциясы, үш логикалық **операция бар**.

Қатыс операциялары:

Теңдікті тексеру (=) Тең еместікті тексеру (! =) Кіші (<)

Кіші не Тең (< =)

Ұлкен (>)

Үлкен не Тең (> =)

Логикалық операциялар - және, немесе, жоқ шарттары түрінде жазылады. Жазылу түрлері:

&& - және (and), || - немесе (or), ! - терістеу (not).

Мысалдар:

```
if (k>50 && j == 24)
```

```
if(k<=2 || k>5)
```

Қатыс және логикалық операциялардың нәтижелері әдеттегідей екеу ақ: ақиқат (true) не жалған (false). Нәтиже ақиқат болса, мәні=1 деп, жалған болса, мәні=0 деп белгіленеді.

Меншіктеу операторы өрнек мәнін айнымалыға меншіктеу операциясының соңына ; таңбасын қоюдан тұрады. Меншіктеу операциясы әдеттегідей тендік (=) таңбасымен белгіленеді. Айнымалы **lvalue** (адрестік шама) деп аталады. Мысалы, $y=3*x+5$; операторындағы у айнымалысы — *lvalue*.

Си тілінде қосарлы меншіктеу операциясы да бар. Мысалы, $y = x = a*b+5$; операторында алдымен $a*b+5$ өрнегінің мәні х айнымалысына меншіктеледі, ал ол у айнымалысына меншіктеледі.

Си тіліне қосымша меншіктеу операциялары да енгізілген:

$+=$, $-=$, $*=$ және $\%=$

Олар - айнымалы мәніне $+$, $-$, $*$ не $\%$ операциясы қосылған өрнекті бастапқы Айнымалыға меншіктеу операциялары. Мұндай операцияларды пайдалану тәсілі төменде көрсетілген. **Бірдей** операциялар:

$m += 10$ және

$m -= 10$ және

$m *= 10$ және

$m \%=10$ және

$m=m/10$ $m=m-10$ $m=m*10$ $m=m\%10$

Мұндағы **$m+=10$** командасы m айнымалысының мәнін 10-ға өсіруді білдіреді, ал **$m=m/10$** командасы оң жақтағы m-нің мәнін орнына қойып, және оған 10-ды қосып, нәтижені m айнымалысына меншіктеуді білдіреді. Олардың ерекшеліктері жоқ, тек бірінші түрде жазылған команда екінші түрде жазылған командадан тезірек орындалады.

Бақылау сұрақтары

1 С++ тілінде өрнек және функцияға түсініктеме?

2 Бүтін тұрақтылар?

3 Символдық тұрақтылар?

4 Арифметикалық операциялар?

Әдебиеттер тізімі

1. Берн Страуструп. Язык программирования С++. Москва: 1999г.

2. Бабэ Б. Просто и ясно о Borland С++; пер. с англ. – СПб.: Питер, 1997.

3. Бад Т. Объектно-ориентированное программирование в действии: пер. с англ. СПб.: Питер, 1997. – 464с.

4. Ирэ П. Объектно-ориентированное программирование с использованием С++; пер. с англ. К.: НИПФ ДиаСофтЛтд, 1995. – 480с.

5. Подбельский В.В. Язык С++. – М.: Финансы и статистика, 1966. – 558с.

6. Шамис В.А. Borland С++ Builder. Программирование на С++; пер. с англ. –М.: БИНОМ.-480с.

7. Шилдт Г. Теория и практика С++. Пер. с англ. – СПб.: БХВ-Петербург, 1996.-416с.

8. Архангельский А.Я. Программирование в С++ Builder 6. –М.: ЗАО «Издательство БИНОМ», 2003.-1152с.

9. Б.Пахомов. С/С++ и Borland С++ 2006. Санкт-Петербург.: БХВ-Петербург., 2006.

10. Т.Павловская. С/С++. Структурное программирование. Питер. 2005.

11. Программирование на С++. Учебное пособие. Под ред. А.Д.Хомоненко. Москва.: Альтекс-А, 2003.