

ЛЕКЦИЯ 4

ОПЕРАТОРЫ

1. Операторы описания

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

2. Операторы выражения

Присваивания:

```
i++;
```

```
i=1, j=1;
```

```
int j=k+i++;
```

Вызов функции:

```
printf ("Вызов функции \"printf\"");
```

Вызов функции "printf"

```
sum (mas, m, n);
```

Поставить

дважды

```
getch (); cin.get ();
```

Пустой оператор:

```
for (; ;);
```

3. Операторы управления

`for, while, do while, if, switch,
continue, break`

Условный оператор

•Сокращенная форма:

`if (выражение) оператор`

•Полная форма:

`if (выражение) оператор1
else оператор2`

```
if (56) printf("56 - true\n");  
if(-90) printf("-90 - true\n");
```

```
if (x>y) max=x;  
    else max=y;
```

```
if(10%2) printf("true\n");  
    else printf("false\n");
```

```
printf("Vvedite x: ");
```

```
scanf("%d", &x);
```

```
if(x>=10) printf("X>=10\n");
```

```
else if(x<=5) printf("X<=5\n");
```

```
    else printf("5<X<10");
```

Как работает следующая программа:

```
if (3<2<1) printf("3<2<1-PRAVLNO!!!");  
else printf("3<2<1-ERUNDA!!! ");
```

3<2<1-PRAVLNO!!!

ПОЧЕМУ???

3<2==0

0<1==1

```
if (2==2 || 2==3) printf("True\n");  
else printf("False\n");
```

```
if (2==3 || 2==2) printf("True\n");  
else printf("False\n");
```

```
if (2==2 && 2==3) printf("True\n");  
else printf("False\n");
```

True

True

False

```
if (3<2 && 2<1)  
if (3<2<1) printf("3<2<1-PRAVLNO!!!");  
printf("3<2<1-PRAVLNO!!!");  
else printf("3<2<1-ERUNDA!!!");  
else printf("3<2<1-ERUNDA!!!");
```

3<2<1-ERUNDA!!!

Важное замечание

Нужно: `if (x==5)` оператор

Пишем: `if (x=5)` оператор

Логическая ошибка!!!

Трудно обнаруживаемая!!!

Лучше писать: `if (5==x)` оператор

т.к., если напишем: `if (5=x)` оператор

Компилятор сообщит об ошибке!!!

1 `if (! (x==y))` оператор

2 `if (x>5 && x<10 || y==0)` оператор

3 `if ((x>5 || x<10) && y==0)` оператор

Пример. Как улучшить алгоритм?

```
int number;  
printf("Введите время года:\t");  
scanf("%d", &number);  
if (number < 1 || number >= 5)  
    printf("Нет такого сезона\n");  
if (1 == number) printf("Весна\n");  
if (2 == number) printf("Лето\n");  
if (3 == number) printf("Осень\n");  
if (4 == number) printf("Зима\n");
```

```
int number;  
printf("Введите время года:\t");  
scanf("%d", &number);  
if(number < 1 || number >= 5)  
    printf("Нет такого сезона\n");
```

```
else if(1 == number) printf("Весна\n");  
else if(2 == number) printf("Лето\n");  
else if(3 == number) printf("Осень\n");  
else printf("Зима\n");
```

Не нужен
последний
if

~~else следует
располагать под
соответствующим if~~

Оператор switch

```
switch (выражение)
{
    case метка1: оператор 1
    case метка2: оператор 2
        . . .
    default: оператор n
}
```

Метка – целочисленное выражение

Если значение выражения не совпадает с метками – выполняется оператор, помеченный default

В вариантах следует использовать оператор break 

```
printf("Vvedite natyralnoe chislo ot 0 do 9: ");
scanf("%u",&n);
switch (n)
{ case 1:
  case 3:
  case 5:
  case 7:
  case 9: {puts("vvedeno Nechetnoe chislo");
           break;
        }
  case 2:
  case 4:
  case 6:
  case 8: {puts("vvedeno Chetnoe chislo");
           break;
        }
  default: puts("Vveden 0");
}
```

Vvedite natyralnoe chislo ot 0 do 9: 0
vvedeno Nechetnoe chislo
vvedeno Chetnoe chislo
Vveden 0

Что будет, если break убрать?

Оператор цикла for

for (выражение1; выражение2; выражение3) оператор

1
`int k;
for (k=1;k<=5;k++)
printf ("%d%5d\n", k, k*k) ;`

1	1
2	4
3	9
4	16
5	25

2
`for (n=1; n<1000; n++) ;`

3
`int n, j;
for (n=15; n>0; n--)
{printf ("%d секунд до старта!\n", n) ;
for (j=1; j<10000000000; j++) ;
system ("cls") ;
}`

4 `for (n=3 ; n<60 ; n+=11)
printf ("%2d\n" , n) ;`

5 `for (ch='a' ; ch<='z' ; ch++)
printf ("Kod ASCII %c = %d \n" , ch , ch) ;`

6 `for (i=1 , j=5 ; i<120 ;)
i*=j ;`

7 `for (i=1 ; ;)
{i++ ;
cout<<i ;
if (i>5) break ;
}`



2 3 4 5 6

8

Реализовать алгоритм:

нажимаем любую клавишу – никаких действий,
нажимаем Q – выход.

```
char c;  
for(;;)  
{printf("\nPress any key, Q to quit:");  
scanf("%c", &c);  
if('Q'==c) break;  
}
```

Оператор цикла while

while (выражение) оператор

```
unsigned n, s=0, i=1;
printf("Input n:");
scanf("%u", &n);
while (i<=n)
{
    s+=i;
    i++;
}
printf("%u", s);
```

Сумма первых n
натуральных
чисел

Эквивалентные операторы *FOR* и *WHILE*.

1

`While (выражение) оператор`

`For (;выражение;) оператор`

2

`For (выр1;выр2;выр3) оператор`

`Выр1;`

`while (выр2)`

`{`

`оператор`

`выр3;`

`}`

Оператор цикла do while

do оператор while (выражение) ;

```
unsigned n, s=0, i=1;  
printf("Input n:");  
scanf("%u", &n);
```

```
do  
{ s+=i;  
  i++;  
}  
while (i<n);
```

```
printf("%u", s);
```

Сумма первых n
натуральных
чисел

```
int i=0;
char ch;
while (1)
{cin>>ch;
  if ( '.'==ch) break;
  i++;
}
cout<<i<<' \n' ;
```

```
ab c d.
4
```

```
int i=0;
char ch;
while (1)
{cin.get(ch);
  if ( '.'==ch) break;
  i++;
}
cout<<i<<' \n' ;
```

```
ab c d.
6
```

`cin.get()` ; или `getch()` ; ждут
нажатия клавиши

При вводе текста с клавиатуры конец файла формируется нажатием клавиш CTRL/Z

```
int i=0;
char ch;
cin>>ch;
while(!cin.fail())
{ i++;
  cin>>ch;
}
cout<<i<<' \n' ;
```

```
int i=0;
char ch;
cin.get()>>ch;
while (!cin.fail())
{ i++;
  cin.get()>>ch;
}
cout<<i<<' \n' ;
```

Aaa
Bb
Ccc
^Z
8

Aaa
Bb
Ccc
^Z
1

Функции `cin.fail()` и `cin.eof()` обнаруживают конец файла

```
int i=0;
char ch;
cin.get()>>ch;
while (!cin.eof())
{
    i++;
    cin.get(ch);
}
cout<<i<<' \n' ;
```

Оператор continue

```
#define D '@'
void main()
{ char s;
  while ( (s=getchar()) !=D)
  {
    if (s=='\n')
      continue;
    putchar(s);
  }
  return;
}
```

BCE