

Лекция 6 Базовые понятия языка программирования Си.

6.1. Си – язык *среднего уровня*: в нем соединены лучшие черты языков высокого уровня с возможностями ассемблера. Как язык среднего уровня Си позволяет манипулировать битами, байтами, адресами, обеспечивая при этом высокую степень переносимости Си-программ.

6.2. *Пример Си-программы: Hello, World!*

6.3. *Структура Си-программы*: Си-программа состоит из одного или нескольких *программных файлов* (модулей, единиц компиляции).

Структура каждого программного файла:

```
объявления глобальных переменных
int main(список параметров); {
последовательность операторов
}
f1(список параметров); {
последовательность операторов
}
f2(список параметров); {
последовательность операторов
}
.....
fN(список параметров); {
последовательность операторов
}
```

Кроме того, программный файл может содержать инструкции препроцессора: **#include**, **#define** и др.

6.4. *Типы данных.*

6.4.1. *Базовые типы данных*: **char** (символьный), **int** (целый), **float** (с плавающей точкой), **double** (двойной точности), **void** (без значения).

6.4.2. *Модификаторы* базовых типов (определяют новые типы данных): **signed**, **unsigned**, **long**, **short**. К типу **int** применимы все модификаторы; к типу **char** – только **signed** и **unsigned**, к типу **double** – только **long**. (в C'99 для типа **double** добавлен модификатор **long long**).

6.5. *Переменные.*

6.5.1. Переменная = тип + имя (*идентификатор*) + значение (оно, вообще говоря, изменяется при выполнении программы); доступ к текущему значению переменной осуществляется по ее имени (либо по указателю). Каждая переменная является *объектом* программы. *Ключевые слова* (в стандарте C89 их 32, в стандарте C99 добавлено еще 5; кроме того «почти ключевым» является слово **main** – см. *дополнение 6.1*) могут использоваться только как ключевые слова, и *не могут быть именами переменных*. Тип переменной определяет интерпретацию операций, операндом которых является эта переменная, и тем самым определяет тип соответствующего выражения.

6.5.2. Объявление переменной (синтаксис): *тип список_переменных*. Например:

```
int i, j, k = 3;
```

При объявлении переменной можно задать ее начальное значение.

6.5.3. Области действия переменных. Переменная может быть объявлена:

(1) внутри функции, или даже части функции – блока¹ (*локальная*); (2) в объявлении функции (*параметр функции*); (3) вне всех функций (*глобальная*).

Областью действия локальной переменной является *блок*, в котором она объявлена. *Областью действия* глобальной переменной является программный файл, в котором она объявлена, начиная со строки файла, в которой содержится ее объявление. В одной области действия нельзя объявлять более одной переменной с одним и тем же именем, а в разных областях действия – можно. Области действия переменных образуют «правильную скобочную структуру». Внутри области действия имя переменной «заслоняет» совпадающие с ним имена переменных из объемлющих областей действия.

6.5.4. Спецификаторы класса памяти: **extern**, **static**, **auto**, **register**.

6.5.5. Квалификатор **const**: объект, объявленный с квалификатором **const**, является *константой* и не может изменять свое значение.

6.5.6. Инициализация переменных при их объявлении; например **char ch1 = 'a'**;

Локальные переменные инициализируются каждый раз при входе в соответствующий блок, глобальные и статические локальные – только один раз в начале работы программы. Неинициализированные локальные переменные до первого присваивания имеют произвольное значение, неинициализированные глобальные и статические локальные – обнуляются компилятором.

6.6. **Константы** (литералы). Символьные константы ('c', '%'); целые константы (100, -34L, 1000U, 999u); константы с плавающей точкой (11.123F, 4.56e-4f), константы двойной точности (1.0, 11.123, 3.1415926L, 6.626068e-34LL); шестнадцатеричные константы (0x80), восьмеричные константы (012); строковые константы ("a", "Hello, World!"); специальные символьные константы (\n, \t, \b и др.).

6.7. **Пример Си-программы.**

```
/* "Магическое" число – простейший вариант */
#include <stdio.h>
#include <stdlib.h>

int main() {
    int magic; /* "магическое" число */
    int guess; /* угаданное число */

    magic = rand() /* генерация "магического" числа */
    /* rand – генератор случайных чисел из библиотеки stdlib */

    printf("Угадай магическое число: ")
    scanf(" %d", &guess);
```

¹ Блок – последовательность операторов, заключенная в фигурные скобки (подробнее см. ниже).

```
if(guess == magic) printf("***Угадал***");  
else printf("***Не угадал***");  
return 0;  
}
```

6.8. Операторы.

- 6.8.1. Условные операторы.
- 6.8.2. Операторы цикла.
- 6.8.3. Операторы перехода.
- 6.8.4. Блок (составной оператор).

6.9. Функции.

- 6.9.1. Объявление функции. Формальные параметры. Возвращаемое значение. Побочный эффект функции. Функции типа **void**.
- 6.9.2. Определение функции
- 6.9.3. Библиотечные функции. Стандартные библиотеки языка Си. Директива **#include** и заголовочные файлы. Препроцессор.
- 6.9.4. Вызов функции. Фактические параметры и способ их передачи (по значению).

6.10. **Сборка Си-программы**: препроцессирование, компиляция, компоновка.

6.11. Пример Си-программы.

```
/* "Магическое" число - чуть более сложный вариант */  
#include <stdio.h>  
#include <stdlib.h>  
int main() {  
    int magic; /* "магическое" число */  
    int guess; /* угаданное число */  
  
    magic = rand() /* генерация "магического" числа */  
  
    printf("Угадай магическое число: ")  
    scanf("%d", &guess);  
  
    if(guess == magic) {  
        printf("***Угадал***");  
        printf("Магическое число равно %d\n", magic);  
    }  
    else {  
        printf("***Не угадал***");  
        if(guess > magic) printf("Слишком большое.\n");  
        else printf("Слишком маленькое.\n");  
    }  
    return 0;  
}
```

}

Дополнение д6.1 Список ключевых слов языка Си.

Д6.1.1. В стандарте *ANSI C'89* определены следующие **ключевые слова**:

| | | | |
|-----------------|---------------|-----------------|-----------------|
| auto | double | int | struct |
| break | else | long | switch |
| case | enum | register | typedef |
| char | extern | return | union |
| const | float | short | unsigned |
| continue | for | signed | void |
| default | goto | sizeof | volatile |
| do | if | static | while |

Д6.1.2. В стандарт *ANSI C'99* добавлены следующие **ключевые слова**: **_Bool**, **_Complex**, **_Imaginary**, **inline**, **restrict**

Д6.1.3. В языке Си различаются верхний и нижний регистры символов (код *ASCII*), так что, например, **else** – ключевое слово, а **ELSE** или **Else** – нет.