

## Лабораторная работа №1

### Разработка программных эмуляций аппаратного обеспечения

Цели и задачи лабораторной работы

- получение навыков программирования на языке C++;
- изучение основных принципов построения программного обеспечения по эмуляции работы аппаратного обеспечения.

### Теоретический материал и задания для получения навыков и умений

В качестве языка разработки программного обеспечения для дисциплины «Аппаратные средства вычислительной техники» был выбран язык C++. Данный язык позволяет реализовывать практически все применяемые на сегодняшний день парадигмы программирования и при этом имеет возможность прямого доступа к аппаратному обеспечению компьютера.

Каждая программа на C++ содержит одну или несколько функций, причем одна из них должна иметь имя *main*. Запуская программу C++, операционная система вызывает именно функцию *main*. Примером простейшей программы на C++ является программа, представленная ниже.

Листинг 1.

```
int main()
{
    return 0;
}
```

Определение функции состоит из четырех элементов:

1. тип возвращаемого значения;
2. имя функции;
3. список параметров;
4. тело функции.

Функция `main` имеет возвращаемый тип `int`, который является встроенным типом языка. Это связано с тем, что единственный оператор, который используется в функции `main` - `return` - используется операционной системой как индикатор результата работы программы. По сложившейся традиции код 0 обозначает успешное завершение программы, а коды отличные от нуля являются кодами ошибок.

### ***Компиляция и выполнение программы***

Написанную программу необходимо откомпилировать. Способ компиляции программы зависит от используемой операционной системы и компилятора.

В качестве основной операционной системы при изучении дисциплины «Аппаратные средства вычислительной техники» используется Linux Mint (либо любой другой deb-based дистрибутив). Для обеспечения одинаковой компиляции программ в различных операционных системах предлагается использовать компилятор *gcc*. Для того чтобы его использовать, создайте файл с именем `lr1.cpp` в директории `/home/%username%/groupname%/student_lastname`, после чего запишите в него код, представленный в листинге 1.

Для того, чтобы скомпилировать команду необходимо выполнить следующую команду:

```
user@host$ gcc lr1.cpp -o lr1 -lstdc++
```

Данная команда выполняет следующие действия:

`-o` - выполняет создание исполняемого файла, имя которого указывается после ключа

`-lstdc++` - данный ключ указывает компилятору, что программа написана именно на языке C++.

### ***Типы данных***

C++ является строго типизированным языком, поэтому тип данных указывает при инициализации переменной. Как и во всех языках, в C++ имеются predefined типы данных.

Арифметические типы данных бывают следующие:

Тип	Значение	Максимальный размер
bool	логический тип	не определен
char	символ	8 бит
wchar_t	двойной символ	16 бит
char16_t	символ Unicode	16 бит
char32_t	символ Unicode	32 бита
short	Целое	16 бит
int	Целое	16 бит
long	Длинное целое	32 бита
long long	Длинное целое	64 бита
float	Число с плавающей точкой	6 значащих цифр
double	Число с плавающей точкой двойной точности	10 значащих цифр
long double	Число с плавающей точкой повышенной точности	10 значащих цифр

За исключением типа `bool`, все остальные типы могут являться как знаковыми (`signed`), так и беззнаковыми (`unsigned`). Если знаковость типа не определена в тексте программы, то она определяется компилятором исходя из аппаратного обеспечения на котором запущена программа, однако в большинстве случаев арифметические типы все-таки по умолчанию являются знаковыми.

### ***Ввод и вывод данных. Структуры следования***

Ввод и вывод данных осуществляется следующим образом:

Листинг 3.

```
#include <iostream>

using namespace std;

int main() {

std::cout<<"Simple output"<<"\n";

return 0;
}
```

В листинге 3 появляется несколько новых определений. В начале программы подключается библиотека для работы со стандартным потоком ввода/вывода, после чего указывается директива `using namespace std`, которая позволяет обращаться напрямую к функциям стандартной библиотеки. Чтобы проверить это, удалите из программы в листинге 3 строку с `using`, после чего попробуйте скомпилировать программу. Затем верните эту строку, и удалите `std::` и проверьте результат.

Ввод данных осуществляется по способу, представленному в листинге 4.

Листинг 4.

```
#include <iostream>

using namespace std;

int main() {

    int x;

    cin>>x;

    x=x*2;

    cout<<"You're doubled number="<<x<<"\n";

    return 0;
}
```

### ***Условный оператор и операторы цикла***

Условный оператор реализуется в C++ следующим образом:

Листинг 5.

```
#include <iostream>

using namespace std;
```

```
int main() {  
  
    int x;  
  
    cin>>x;  
  
    x=x*2;  
  
    if (x<50)  
    {  
        cout<<"less 50";  
    }  
    else  
    {  
        cout<<"More 50 or equal";  
    }  
  
    return 0;  
}
```

Оператор цикла с перечислением реализуется следующим образом:

Листинг 6.

```
#include <iostream>  
  
using namespace std;  
  
int main() {  
  
    for (int i=0;i<10;i++){  
        cout<<i<<"\n";  
    }  
  
    return 0;  
}
```

Оператор цикла с предусловием реализуется следующим образом (выполняется то же самое, что и листингом выше):

Листинг 7.

```
#include <iostream>

using namespace std;

int main() {

    int i=0;

    while (i<10)
    {
        cout<<i<<"\n";
        i++;
    }

    return 0;
}
```

### ***Некоторые функции***

Для решения некоторых математических задач может использоваться библиотека *cmath*, которая подключается аналогично библиотеке *iostream*. Некоторые функции этой библиотеки представлены в таблице.

<code>abs( a )</code>	модуль или абсолютное значение от $a$	<code>abs(-3.0)=3.0;abs(5.0)= 5.0</code>
<code>sqrt(a)</code>	корень квадратный из $a$ , причём $a$ не отрицательно	<code>sqrt(9.0)=3.0</code>
<code>pow(a, b)</code>	возведение $a$ в степень $b$	<code>pow(2,3)=8</code>
<code>ceil( a )</code>	округление $a$ до наименьшего целого, но не меньше чем $a$	<code>ceil(2.3)=3.0;ceil(-2.3)=-2.0</code>
<code>floor(a)</code>	округление $a$ до наибольшего целого, но не больше чем $a$	<code>floor(12.4)=12;floor(-2.9)=-3</code>
<code>fmod(a, b)</code>	вычисление остатка от $a/b$	<code>fmod(4.4, 7.5) = 4.4;fmod( 7.5, 4.4) = 3.1</code>
<code>exp(a)</code>	вычисление экспоненты $e^a$	<code>exp(0)=1</code>
<code>sin(a)</code>	$a$ задаётся в радианах	
<code>cos(a)</code>	$a$ задаётся в радианах	

### Задания для самостоятельного выполнения

С помощью языка C++ решите следующие задачи:

1. Найти площадь кольца по заданным внешнему и внутреннему радиусам.
2. Даны катеты прямоугольного треугольника. Найти его периметр.
3. Известны координаты на плоскости двух точек. Составить программу вычисления расстояния между ними.
4. Даны основания и высота равнобедренной трапеции. Найти периметр трапеции.
5. Треугольник задан координатами своих вершин. Найти периметр и площадь треугольника.
6. Даны объёмы и массы двух тел из разных материалов. Материал какого из тел имеет большую плотность?
7. Год является високосным, если его номер кратен 4, однако из кратных 100 високосными являются лишь кратные 400, например, 1700, 1800 и 1900 — не-високосные года, 2000 — високосный. Дано натуральное число  $n$ . Определить, является ли високосным год с таким номером.
8. Дано пятизначное число. Найти число, получаемое при прочтении его цифр справа налево.

9. Дано шестизначное число. Найти сумму его цифр. Величины для хранения всех шести цифр числа не использовать.
10. Дано натуральное число. Определить номер цифры 8 в нем, считая от конца числа. Если такой цифры нет, ответом должно быть число 0, если таких цифр в числе несколько — должен быть определен номер самой левой из них.
11. Дано натуральное число. Определить, сколько раз в нем встречается минимальная цифра (например, для числа 102 200 ответ равен 3, для числа 40 330 — 2, для числа 10 345 — 1).
12. Найти 20 первых натуральных чисел, делящихся нацело на 13 или на 17 и находящихся в интервале, левая граница которого равна 500.
13. Дано натуральное число. Если в нем есть цифры  $a$  и  $b$ , то определить, какая из них расположена в числе правее. Если одна или обе эти цифры встречаются в числе несколько раз, то должны быть учтены самые правые из одинаковых цифр.
14. Найти наибольший общий делитель двух заданных натуральных чисел, используя алгоритм Евклида.
15. Найти наименьшее общее кратное двух заданных натуральных чисел.

## Содержание и оформление отчета

В отчет по лабораторной работе включается следующая информация:

1. Название лабораторной работы
2. Основные положения теоретического материала
3. Результаты выполнения самостоятельных заданий в виде листингов программ.