

Лабораторная работа 6. Управляющие операторы в языке C++. Часть 1

1. ЦЕЛЬ РАБОТЫ

Изучить возможности использования простейших управляющих конструкций в программах на языке C++.

2. КРАТКИЕ ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

Условный оператор if... else. Условный оператор реализует структуру выбора. Он имеет такой вид:

if (условие) оператор1 **else** оператор2

Если *условие* оценивается как истинное, выполняется *оператор1*, если как ложное, выполняется *оператор2*. Простейший пример:

```
if (a > b) max_ab = a; else max_ab = b;
```

Пример 1 программы с использованием инструкции if. При выполнении этой программы пользователю предлагается ввести два разных числа, а затем сообщается результат их сравнения.

```
#include <iostream>
using namespace std;

int main()
{int a,b;
  cout << "Enter first number: ";
  cin >> a;
  cout << "Enter second number: ";
  cin >> b;

  if (a > b) cout << "a > b \n";
  else cout << "b >= a \n";

  return 0;
}
```

Вместо одиночного оператора всегда можно подставить *блок* из нескольких операторов. *Блок* – это логически связанная группа программных инструкций, которая обрабатывается как единое целое. В C++ программный блок создается путем записи группы инструкций внутри фигурных скобок. **Пример 2:**

```
#include <iostream>
using namespace std;

int main()
{int a,b;
  cout << "Enter first number: "; cin >> a;
  cout << "Enter second number: "; cin >> b;

  if (a > b)
  {cout << "a > b \n";
```

```

    cout << "Ih raznost' ravna " << a-b;
}
else
{cout << "b >= a \n";
  cout << "Ih raznost' ravna " << b-a;
}

return 0;
}

```

Условие оператора `if` может быть сколь угодно сложным выражением. Причем тип выражения в операторе `if` не обязательно должен быть `bool`. Главное, чтобы результат вычисления выражения можно было интерпретировать как значение ИСТИНА или ЛОЖЬ. Так, например, нуль автоматически преобразуется в `false`, а все ненулевые значения – в `true`. Таким образом, любое выражение, которое дает в результате нулевое или ненулевое значение можно использовать для управления инструкцией `if`.

В условиях в операторе `if` могут использоваться логические операции (логические AND и OR) и операции отношения (меньше, больше и т.д.). Операции отношения служат для сравнения (больше — меньше) или проверки на равенство двух числовых операндов. Операции возвращают “логическое” значение, т. е. ненулевое целое в случае, если условие отношения удовлетворяется, и нулевое в противном случае. Перечень логических операций и операций отношения приведен в таблице «Операции языка C++» в конспекте к 1 лекции.

Примеры условий

Выражение	Пояснение
<code>x != 10</code>	Значение <code>x</code> не равно 10
<code>(x >= 0) && (y >= 0)</code>	Значения переменных <code>x</code> и <code>y</code> неотрицательны
<code>(x >= -10) && (x <= 10)</code> <code> (count == 5)</code>	<code>x ∈ [-10; 10]</code> или значение переменной <code>count</code> равно 5

Для создания консольного приложения на языке C++ в интегрированной среде Visual Studio 20XX необходимо выбрать в меню **File** следующую последовательность опций: **New->Project-> ->Visual C++ -> Win32 Console Application**. После этого определяем имя проекта в поле **Name** и местонахождение каталога с файлами проекта в поле **Location**. После нажатия **OK** в окне **New Project** появится окно **Win32 Application Wizard**, в котором необходимо выбрать пункт **Application Settings**. В правой половине окна появится группа селективных кнопок **Application Type** для указания типа приложения и две группы кнопок-переключателей. Для консольного проекта необходимо выбрать селективную кнопку **Console application** и переключатель **Empty Project**, а затем нажать кнопку **Finish**.

3. ЗАДАНИЕ НА РАБОТУ

1. Проверить работу программ, приведенных в кратких теоретических положениях.
2. Разработать самостоятельно приложения для решения трех задач по своему варианту.

4. ПОРЯДОК ВЫПОЛНЕНИЯ РАБОТЫ

Задание 1. Ознакомьтесь с теоретическим материалом, приведенным в пункте «Краткие теоретические положения» данных методических указаний и с конспектом лекций по данной теме.

Задание 2. Создайте новое консольное приложение, как указано выше. Наберите текст программы из примера 1. Проверьте работу приложения. Покажите результат преподавателю.

Задание 3. Проверьте работу программы из примера 2. Измените программу (Пример 2) таким образом, чтобы в ней при равенстве значений переменных *a* и *b* выводилось какое-либо соответствующее сообщение.

Задание 4. Далее необходимо разработать три приложения по своему варианту (см. таблицу с вариантами). В каждом варианте – три задачи.

Покажите результаты преподавателю. Оформите отчет по работе.

5. ОФОРМЛЕНИЕ ОТЧЕТА

Отчет по работе должен содержать:

- название и цель работы;
- номер варианта;
- для задач по своему варианту – текст задачи, текст кода программы, схема алгоритма программы, результаты выполнения разработанной программы для разных наборов исходных данных; расчеты, выполненные вручную для тех же наборов исходных данных.

6. БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Шилдт Г. С++: базовый курс, 3-е издание. : Пер. с англ. – М.: «Издательский дом «Вильямс», 2005. – 624 с.
2. Пахомов Б.И. С/C++ и MS Visual C++ для начинающих. – СПб.: БХВ-Петербург, 2008. – 624 с.

7. ПРИМЕРЫ ЗАДАНИЙ

1. Даны целочисленные координаты трех вершин прямоугольника, стороны которого параллельны координатным осям. Найти координаты его четвертой вершины.

2. Заданы координаты левой верхней и правой нижней вершин прямоугольника (x_1, y_1), (x_2, y_2). Определить площадь части прямоугольника, расположенной в первой координатной четверти.
3. Вычислить количество корней и сами корни квадратного уравнения с заданными коэффициентами A, B, C . Если действительных корней нет, сообщить об этом.
4. Известны площади круга и квадрата. Уместится ли круг в квадрате? Уместится ли квадрат в круге?