

Лабораторная работа №2
Курс "Основы программирования"
Тема: Программирование ветвления и
циклов на языке C++

Цель: Научиться применять операторы ветвления и циклов на языке C++.

Контрольные вопросы:

1. Приведите по три примера задач, которые нужно решить с помощью операторов ветвления и с помощью операторов цикла.

Оператор ветвления:

1. Нахождение максимального числа из трех
2. Определение четности или нечетности числа
3. Решение квадратного уравнения

Операторы цикла:

1. Нахождение суммы последовательности до последнего элемента массива с клавиатуры.
2. Вывод на экран значений текущего момента, заданного в цикле.
3. Запись данных в массив

2. Чем отличаются составные логические выражения от простых? Приведите по два примера простых и составных выражений на языке С++.

Простое логическое выражение состоит из одного высказывания и не содержит логической операции.

Сложное логическое выражение содержит высказывания, объединенные логическими операциями.

На основании простых высказываний могут быть построены составные.

Простое:

1. $A > B, B > C$

2. $a \neq 0$

Составное:

1. $(A > B) \&\& (B > C)$; 2. $(a = 1) \parallel (b > 5)$

3. Запишите строку кода $\text{int } x = n \% 3 == 0$
 $n/3 : n * 3$; кодом с оператором `if`

`int x;`

`x = n % 3;`

`if (n % 3 == 0) { x = n / 3; }`

`else { x = n * 3; }`

4. Переменные каких типов можно использовать в качестве селекторов `switch`?

В качестве селекторов `switch` можно использовать типы данных: `int`, `char`, `byte`, `short`, `long`, `string`.

5. Что делают операторы `break` и `continue`?

Оператор `break` позволяет досрочно прервать цикл. `break` прерывает текущий цикл и продолжает выполнение следующих выражений. Если используется несколько вложенных циклов, `break` прерывает внутренний цикл и продолжает выполнять выражения, следующие за блоком `break` может использоваться в циклах `for` и `while`.

Оператор `continue` возвращает управление к началу цикла. Т.е. `continue` позволяет пропустить оставшиеся выражения в цикле и перейти к следующей строке.

6. Запишите цикл, который считает сумму всех четных чисел в диапазоне от 0 до 20, в виде оператора `for` с условием.

```
int sum = 0;  
for (int i = 1; i <= 20; sum += i + 1)  
{ }
```

7. В чем отличие оператора `do while` от оператора `while`?

Отличие заключается в том, что цикл `while` будет выполняться пока условие, указанное в круглых скобках, является истиной. В то время, как цикл `do while`, хоть, внешне похож на цикл `while`, при выполнении цикла один проход цикла будет выполнен независимо от условия.

Вывод: В ходе лабораторной работы я закрепила знание по применению операторов ветвления и цикла.