Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования

Томский государственный университет систем управления и радиоэлектроники (ТУСУР)

Кафедра экономической математики, информатики и статистики (ЭМИС)

ОПЕРАТОРЫ ВЕТВЛЕНИЯ И ЛОГИЧЕСКИЕ ВЫРАЖЕНИЯ В ЯЗЫКЕ С/С++

Отчёт по лабораторной работе по дисциплине «Введение в программирование»

Студент гр. 543-1

/А.Е. Мухамеджан

“\_\_\_\_”\_\_\_\_\_\_\_\_\_\_\_2023

Руководитель доцент

кафедры ЭМИС

\_\_\_\_\_\_\_\_ Е.А. Шельмина

“\_\_\_”\_\_\_\_\_\_\_\_\_\_\_\_2023г.

Томск 2023

Лабораторная работа №3

Операторы ветвления и логические выражения в языке С/С++

Цель работы: изучить операторы условия, операции отношения, логические операции и выражения языка программирования Си.

Теоретические сведения

Операции сравнения применяются при работе с двумя операндами и возвращают true (1), если результат сравнения – истина, и false (0), если результат сравнения – ложь. В языке Си определены следующие операции сравнения: < (меньше), <= (меньше или равно), > (больше), >= ( больше или равно), != ( не равно), = = (равно).

Логические операции работают с операндами скалярных типов и возвращают результат булева типа. Существует три логические операции:

* ! – отрицание, или логическое НЕ
* && - логическое И
* || - логическое ИЛИ.

Приоритет операций в С/С++ представлен на рисунке 1.

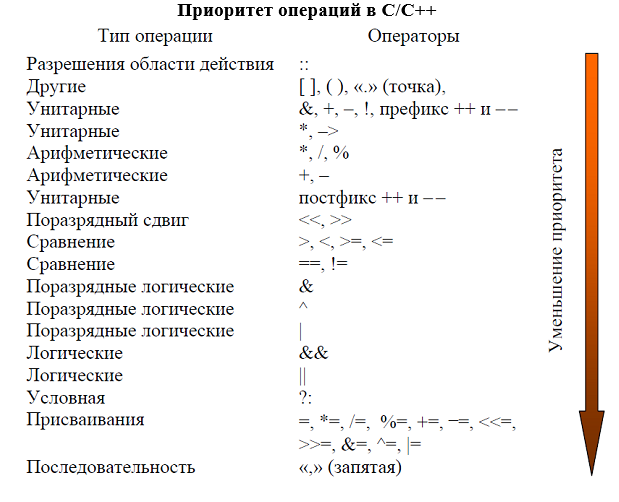


Рисунок 1 – Скриншот таблицы приоритета операций в C/C++

Оператор if ….. else

Когда требуется не просто выполнять операции последовательно (линейный процесс), а изменять ход вычислительного процесса в зависимости от некоторых условий, используется **ветвление по условию**

Формат оператора if … else :

*БЛОК 1;*

*if ( условие )*

*{ БЛОК 2; }*

*else*

*{ БЛОК 3; }*

*БЛОК 4;*

Блок-схема оператора представлена на рисунке 2.

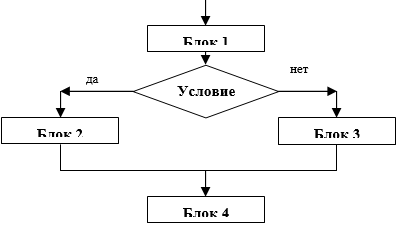


Рисунок 2 – Скриншот блок-схемы оператора условия

Правило выполнения: сначала выполняется БЛОК 1, затем, если указанное условие истинно, то выполняется БЛОК 2, если ложно, то выполняется БЛОК 3. Затем выполняется БЛОК 4. Т.е. в зависимости от условия выполняется либо БЛОК 2, либо БЛОК 3.

Итак, все зависит от выполнения или невыполнения условия. Обычно условия могут состоять из переменных, выражений и функций. Единственное требование состоит в том, что сравнивать можно сравнимое.

Любое выражение, в том числе логическое, всегда имеет значение. Если выражение ложно, то оно равно 0. Истинным является выражение, не равное 0 (положительные, отрицательные, целые, дробные).

Для записи условий применяются специальные знаки представленные на рисунке 3.



Рисунок 3 – Скриншот таблицы специальных знаков для записи условий

Оператор - переключатель *switch( )*

Для тех случаев, когда требуется выбрать действия не из двух вариантов, а из нескольких, предназначен оператор переключатель *switch*.

Конечно, можно было бы использовать несколько операторов *if … else*, но оператор *switch* оказывается намного удобнее. Он позволяет обработать сразу несколько сравнений по условию «выражение равно значению».

Формат:

*switch ( параметр-выражение )*

*{*

*case 1\_значение: { блок 1 }*

*case 2\_ значение: { блок 2 break ; }*

*case 3\_ значение: { блок 3 break ; }*

*case 4\_значение:*

*case 5\_ значение:*

*case 6 \_значение: { блок 4 break ; }*

*case 7 \_значение: { блок 5 }*

*case 8 \_значение: { блок 6 break ; }*

*и т.д.*

*default : { блок 7 }*

*}*

Значения 1-8 – константные выражения целочисленного типа или char, которые сравниваются с параметром-выражением. В качестве параметра может использоваться только переменная целого или символьного типа или целочисленное выражение, содержащее переменные данного типа.

Правило выполнения: Выполняется блок, помеченный значением, которое оказывается равным параметру-выражению. А следом выполняются все последующие блоки, пока не встретится специальный оператор break. Его смысл состоит в том, что он прекращает выполнение главного оператора, внутри которого он находится.

Правило выполнения. (По примеру)

Если параметр равен 1 значению, то выполняются блок 1 и блок 2.

Если параметр равен 2 значению, то выполняется блок 2.

Если параметр равен 3 значению, то выполняется блок 3.

Если параметр равен 4 значению,

или 5 значению,

или 6 значению, то выполняется блок 4.

Если параметр равен 7 значению, то выполняется блок 5 и блок 6.

Если параметр равен 8 значению, то выполняется блок 6.

и т.д.

Если параметр не равен ни одному из заданных значений, то выполняется раздел default т.е. блок 7.

Ход работы

Задание 1. Разработать блок-схему и написать программу на языке С/C++ для решения следующей задачи: вычислить значение функции. Значение x вводить с клавиатуры, используя оператор ***scanf( )***. Если x не принадлежит вышеприведенным числовым интервалам, то вывести на экран соответствующее сообщение. В противном случае вывести значение y, используя функцию ***printf( ).***

Условие к заданию №1 представлено на рисунке 4.

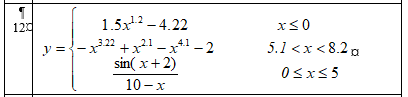


Рисунок 4 – Условие к заданию №1

Блок-схема к заданию №1 представлена на рисунке 5.

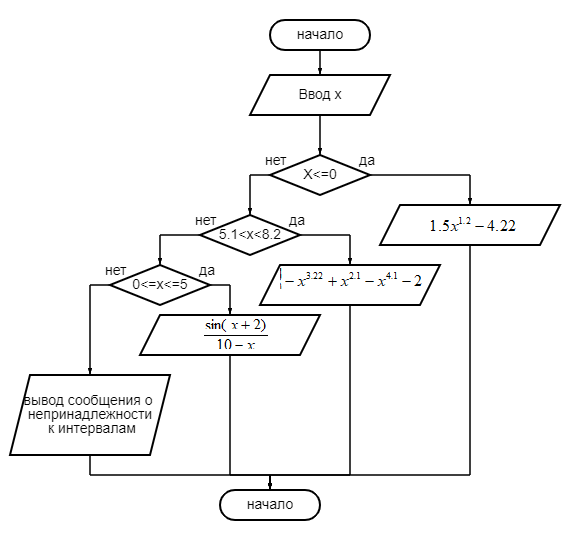


Рисунок 5 - Блок-схема к заданию №1

Программа к заданию №1 представлена на рисунке 6.

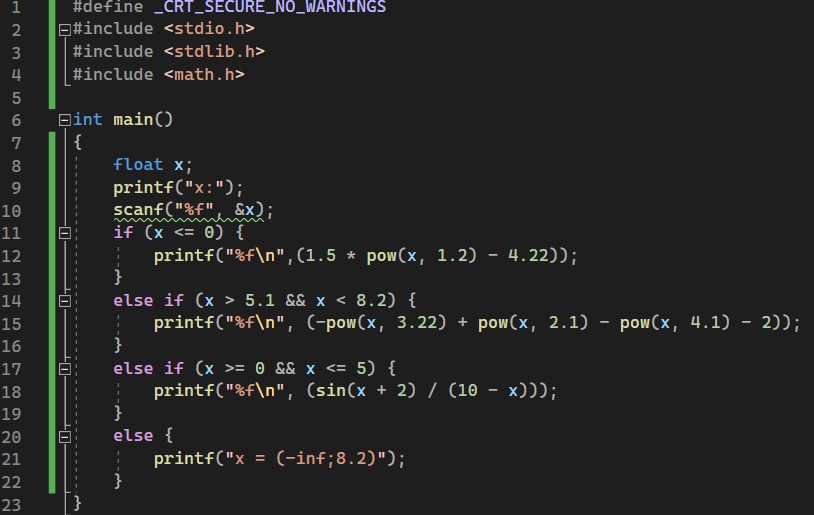


Рисунок 6 – Программа к заданию №1

Вывод программы к заданию №1 представлен на рисунке 7.

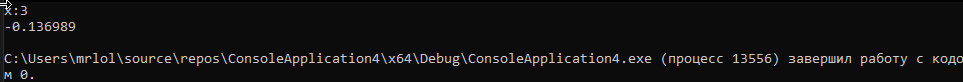


Рисунок 7 – Вывод программы к заданию №1

Задание 2. Написать программу вычисления значения функции y в зависимости от введенного с клавиатуры значения переменной i, используя оператор switch. Необходимые данные вводить по мере необходимости.

Условие к заданию №2 представлено на рисунке 8.

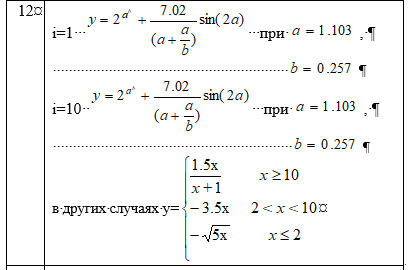


Рисунок 8 – Условие к заданию №2

Программа к заданию №2 представлена на рисунке 9.

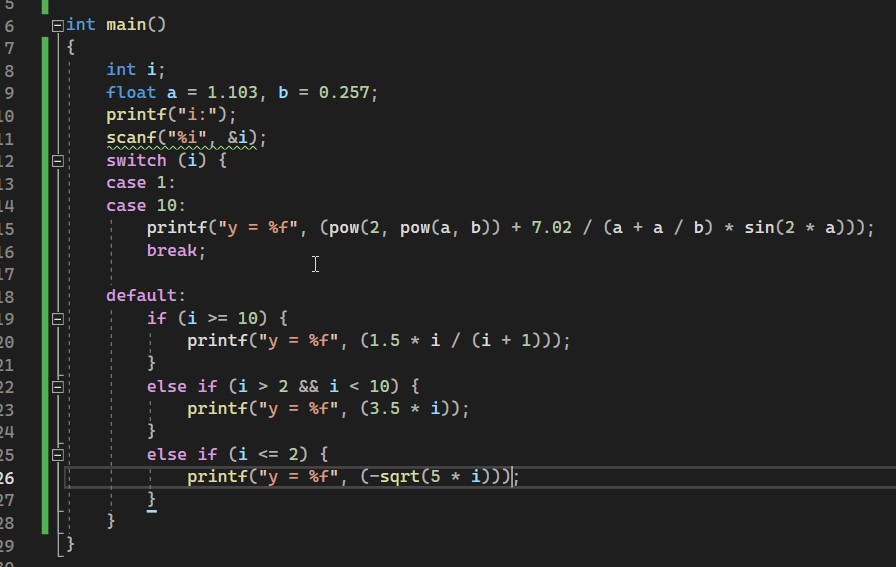


Рисунок 9 – Программа к заданию №2

Вывод программы к заданию №2 представлен на рисунке 10.

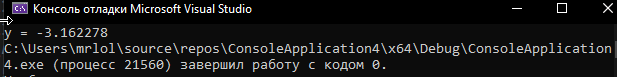


Рисунок 10 – Вывод программы к заданию №2

Задание 3. Разработать блок-схему и написать программу на С/C++ для решения задачи согласно варианту.

Условие к заданию №3 представлено на рисунке 11.

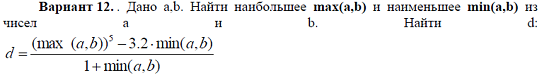


Рисунок 11 - Условие к заданию №3

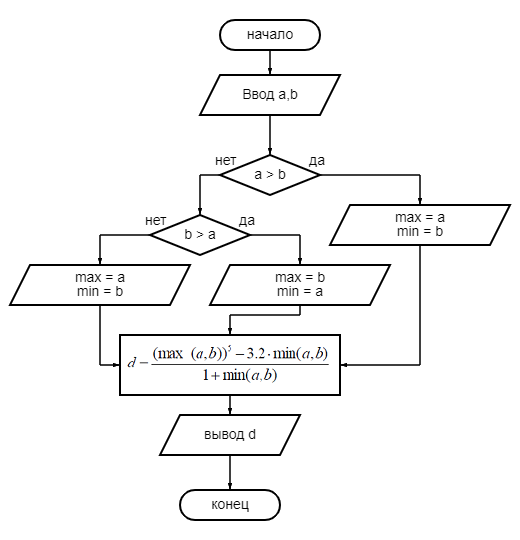
Блок-схема к заданию №3 представлена на рисунке 12 .

Рисунок 12 - Блок-схема к заданию №3

Программа к заданию №3 представлена на рисунке 13.



Рисунок 13 – Программа к заданию №3

Вывод программы к заданию №3 представлен на рисунке 14.

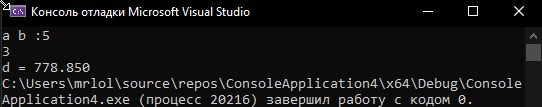


Рисунок 14 – Вывод программы к заданию №3

Задание 4. Написать программу на С/C++ для решения задачи согласно варианту. Использовать оператор **switch**.

Условие к заданию №4 представлено на рисунке 15.



Рисунок 15 - Условие к заданию №4

Программа к заданию №4 представлена на рисунке 16.

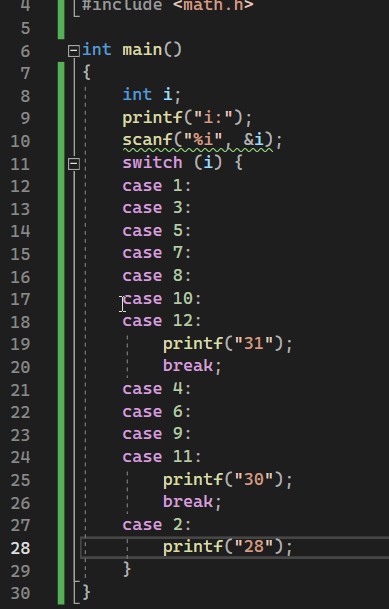


Рисунок 16 – Программа к заданию №4

Вывод программы к заданию №4 представлен на рисунке 17.

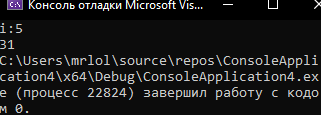


Рисунок 7 – Вывод программы к заданию №1

Задание 5. Для решения приведенных ниже задач написать программу на языке С/С++ с использованием логического типа ***bool***.

Условие к заданию №5 представлено на рисунке 18.

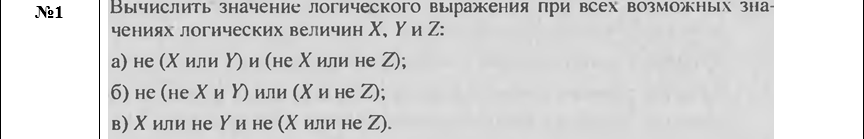


Рисунок 18 - Условие к заданию №5

Программа к заданию №5 представлена на рисунке 19.

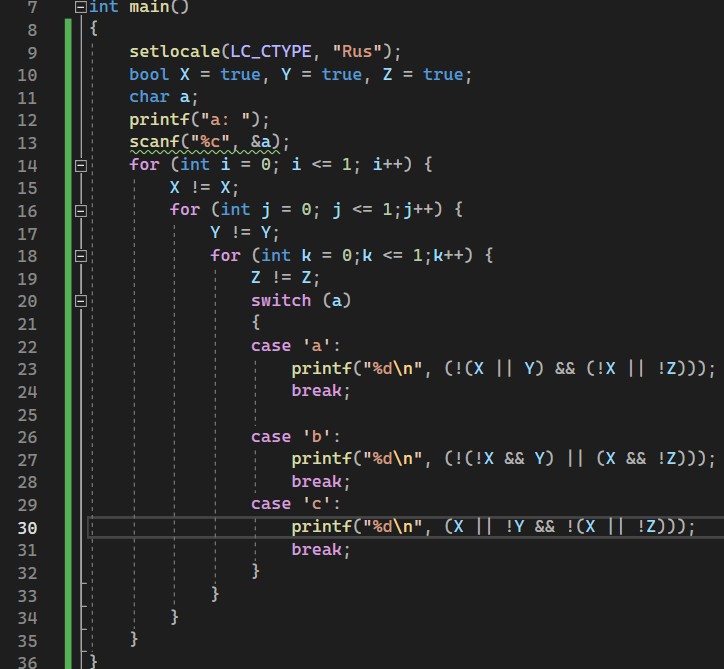


Рисунок 19 – Программа к заданию №5

Вывод программы к заданию №5 представлен на рисунке 20.



Рисунок 20 - Вывод программы к заданию №5

Вывод: в ходе выполненной лабораторной работы были изучены операторы условия, операции отношения, логические операции и выражения языка программирования Си.