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 ЦИКЛА В ЯЗЫКЕ СИ

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 №4

Операторы цикла в языке Си

Цель работы: изучить операторы цикла языка программирования Си.

Теоретические сведения

Циклические вычислительные процессы

Очень часто в программировании бывают случаи, когда требуется многократное повторение одних и тех же действий. Для того, чтобы не было необходимости в многократном повторении одинаковых фрагментов кода программы, предусмотрены специальные операторы циклического повторения выделенного фрагмента программы.

Существует несколько различных операторов, предназначенных для организации циклов:

* *операторы цикла с предусловием,*
* *оператор цикла с постусловием.*

Циклы с предусловием, когда решение об очередном выполнении принимается перед началом его выполнения.

Для организации цикла необходимо:

* *Выбрать переменную, которая, меняясь, будет менять значение условного выражения, и проинициализировать ее. Такая переменная называется управляющей переменной цикла.*
* *Выбрать условное выражение, обеспечивающее повторение тела цикла или прекращение его выполнения.*
* *Определить, какие действия должны выполняться при каждом повторении, т.е. запрограммировать тело цикла.*
* *Внутри цикла произвести коррекцию (изменение) управляющей переменной цикла.*

При этом инициализация управляющей переменной должна быть выполнена до цикла (вне его).

Оператор цикла *while* предназначен для организации цикла с предусловием. Число повторов определяется тем, выполнено или нет условие, которое стоит в начале (заголовке) цикла.

Блок-схема такого цикла представлена на рисунке 1.

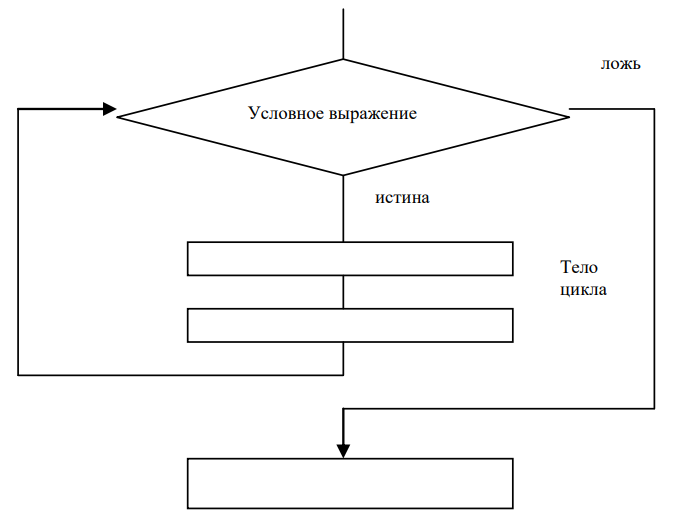


Рисунок 1 – Скриншот блок-схемы цикла с предусловием.

Формат:

*while ( условное выражение )*

*тело цикла;*

Правило выполнения: Цикл повторяется до тех пор, пока выполняется заданное условие, т.е. условие истинно (не равно 0).

Второй оператор цикла с предусловием оператор *for* предназначен для организации цикла с заданным числом повторов. Число повторов задается в параметрах цикла. Циклы данного типа имеют так называемый счетчик. Требуется задать его начальное и конечное значения и шаг изменения. В отличие от предыдущих операторов цикла, где инициализация, проверка условия и коррекция выполняется в разных местах, в операторе *for* они собраны в одном месте.

Формат оператора *for*:

*for (инициализация; условие; коррекция)*

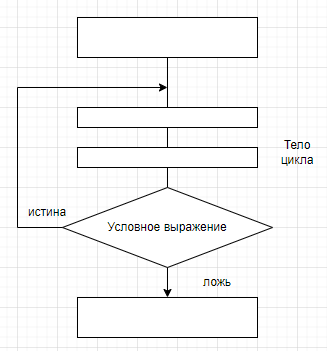
*тело цикла;*

Правило выполнения:

* *Инициализация выполняется только один раз (при первом вхождении в цикл).*
* *Проверка условного выражения. Если условие истинно, то к п.3, иначе – выход из цикла.*
* *Выполняются операторы тела цикла.*
* *Выполняется коррекция, после этого переходим к пункту 2, и так до тех пор, пока условие не станет ложным. Тогда управление передается вниз, на следующий за оператором for оператор.*

Циклы бывают с постусловием, когда решение об очередном выполнении принимается после каждой итерации (выполнения) тела цикла. В этом случае операторы тела цикла выполняются хотя бы один раз.

Блок-схема такого цикла представлена на рисунке 2.

Рисунок 2 – Скриншот блок-схемы цикла с постусловием

Оператор цикла *do … while* предназначен для организации цикла с постусловием. Число повторов определяется тем, выполнено или нет условие, которое стоит в конце цикла.

Отличие от цикла типа *while* состоит в том, что у цикла *while* сначала проверяется условие, а затем выполняется очередной повтор тела цикла (т.е. возможен случай, когда тело цикла не выполняется ни разу), а у цикла типа *do … while* сначала выполняется тело цикла, а потом проверяется условие и в зависимости от него начинается повтор или нет ( т.е. тело цикла всегда выполняется хотя бы 1 раз).

Формат:

*do*

*{*

*тело цикла;*

*}*

*while ( условное выражение );*

Правило выполнения: Цикл повторяется до тех пор, пока выполняется заданное условие. Правила написания условий, такие же, как и в операторе *if … else*.

Типичный пример циклического процесса — вычисление конечных сумм. При определении сумм многократно вычисляется выражение, стоящее под знаком суммы и складывается с предыдущей частичной суммой. Вычисления производятся до тех пор, пока не будут сложены выражения под знаком суммы для всех значений изменяющейся переменной.

Например, необходимо составить программу, вычисляющую значение суммы:

.

Прежде чем вычислять выражение под знаком суммы необходимо определить начальное значения суммы S = 0 и начальное значение параметра цикла i = 1.

Затем вычисляется выражение под знаком суммы для i = 1 и складывается с предыдущим значением суммы S. При этом в переменной S содержится i -я частичная сумма. После этого значение I увеличивается на единицу и проверяется, не стало ли оно больше 10. Если значение I меньше или равно 10, то вычисляется следующая частичная сумма. В противном случае вычисление суммы будет закончено. Конечное значение суммы S выводится на экран.

Так как количество повторений заранее известно, то целесообразно использовать оператор цикла for, включающий в себя блоки инициализации, проверки условия продолжения цикла, изменения переменной цикла после выполнения операторов тела цикла. На рисунке 3 показана простейшая блок-схема вычисления суммы при помощи цикла *for*. В блоке цикла указывается имя параметра цикла, его начальное и конечное значения, шаг изменения (если шаг равен 1 его можно не указывать).

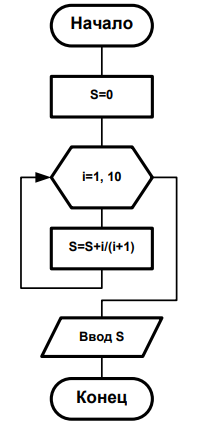


Рисунок 3 – Скриншот блок-схемы вычисления суммы при помощи цикла *for*

Ход работы

Задание 1. Написать программу на языки Си для вычисления конечной суммы, результаты расчета представить в виде таблицы.

Условие к заданию 1 представлено на рисунке 4.

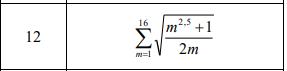


Рисунок 4 – Скриншот условия к заданию №1

Программа к заданию №1 представлена на рисунке 5.

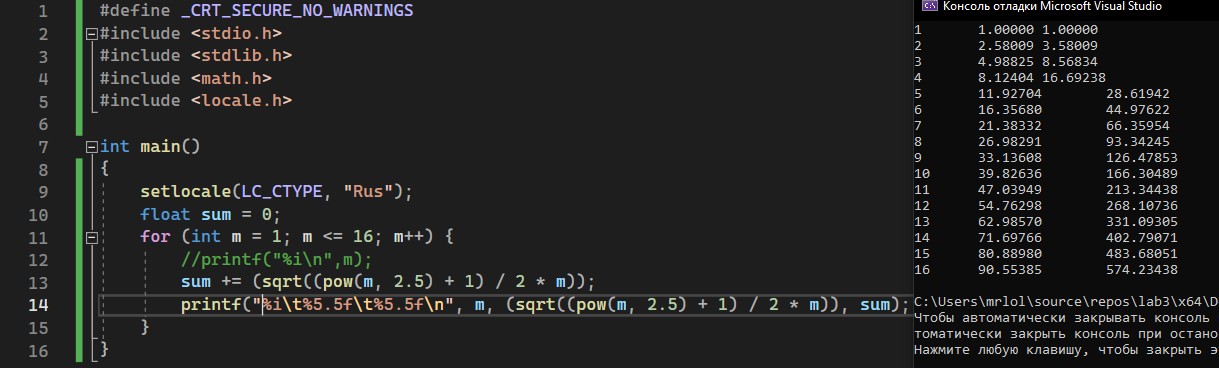


Рисунок 5 – Скриншот программы к заданию №1

Вывод программы представлен на рисунке 6.

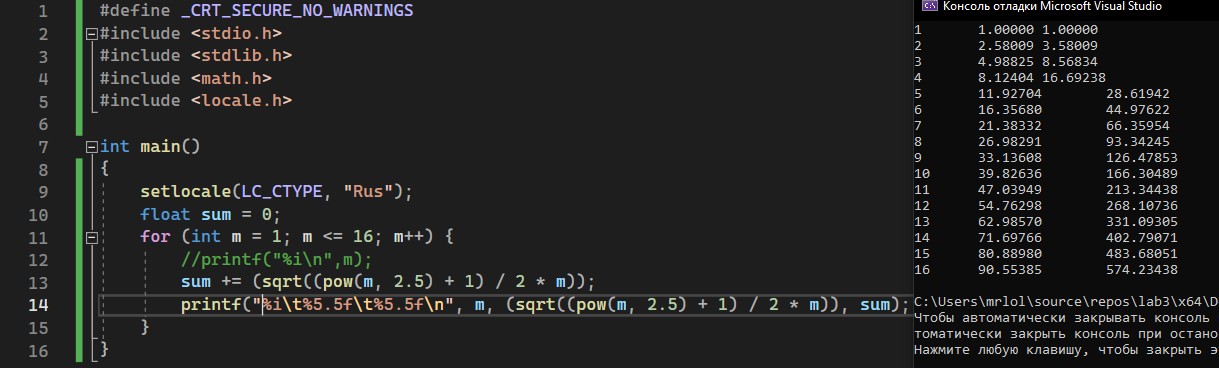


Рисунок 6 – Скриншот вывода программы к заданию №1

Задание 2. Написать программу на языки Си для вычисления бесконечной суммы с заданной точностью.

Условие к заданию №2 представлено на рисунке 7.



Рисунок 7 – Скриншот условия к заданию №2

Программа к заданию №2 представлена на рисунке 8.

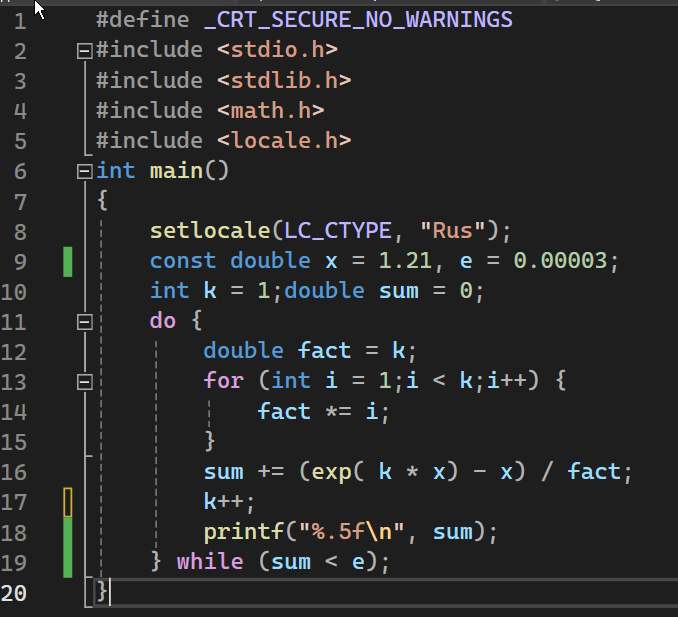


Рисунок 8 – Скриншот программы к заданию №2

Вывод программы представлен на рисунке 9.

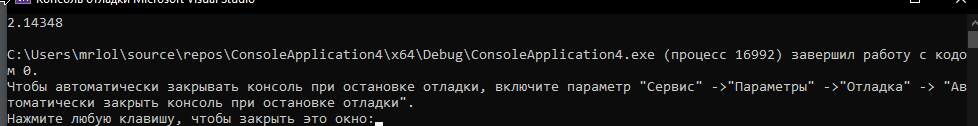


Рисунок 9 – Скриншот вывода программы к заданию 2

Задание 3. Вычислить значения функции f(x) на отрезке [a;b] с шагом h .

Условие к заданию №3 представлено на рисунке 10.



Рисунок 10 – Скриншот условия к заданию №3

Программа к заданию №3 представлена на рисунке 11.

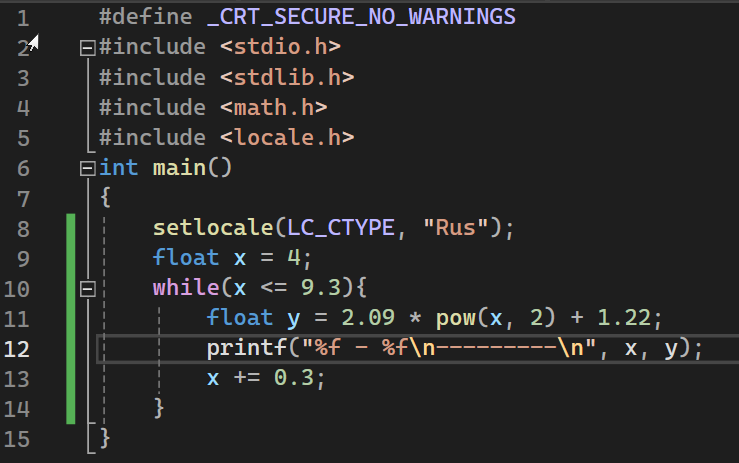


Рисунок 11 – Скриншот программы к заданию №3

Вывод программы представлен на рисунке 12.

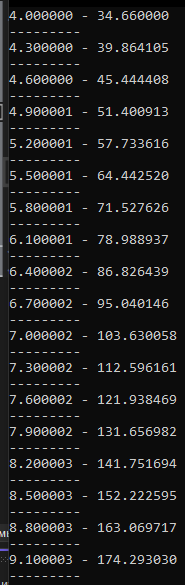


Рисунок 12 – Скриншот вывода программы к заданию №3

Задание 4. Дано натуральное n и действительное x . Вычислить значение суммы.

Условие к заданию №4 представлено на рисунке 13.



Рисунок 13 – Скриншот условия к заданию №4

Программа к заданию №4 представлена на рисунке 14.

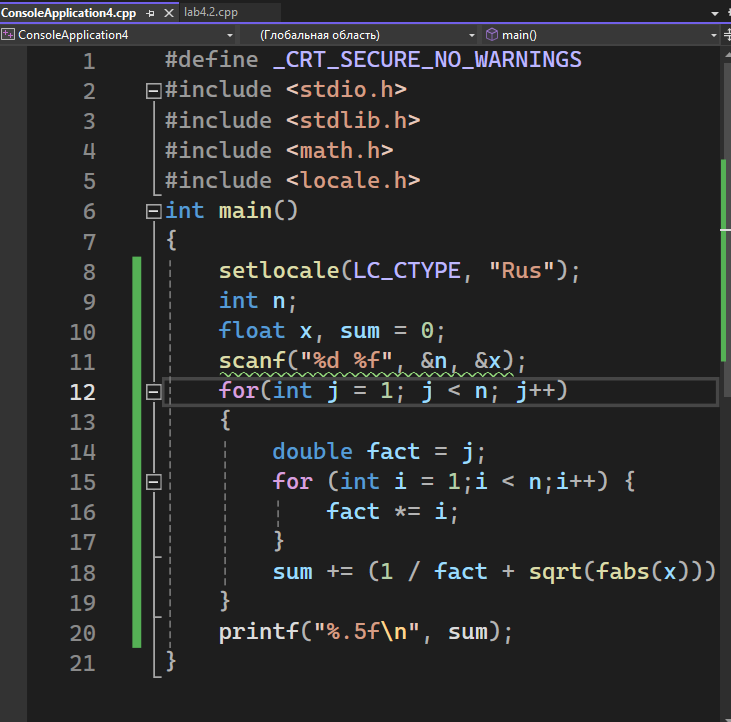


Рисунок 14 – Скриншот программы к заданию №4

Вывод программы представлен на рисунке 15.

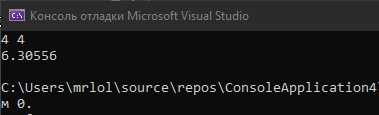


Рисунок 15 – Скриншот вывода программы к заданию №4

Вывод: в ходе лабораторной работы №4 изучены операторы цикла языка программирования Си, выполнены предоставленные задания.