**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**



**МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ВЫСШАЯ ШКОЛА ПЕЧАТИ И МЕДИАИНДУСТРИИ**

***Институт Принтмедиа и информационных технологий***

***Кафедра Информатики и информационных технологий***

**направление подготовки**

**09.03.02 «Информационные системы и технологии»**

**ЛАБОРАТОРНАЯ РАБОТА № \_12\_**

**Дисциплина:** Введение в программирование

**Выполнил(а): студент(ка) группы \_191-726\_\_**

\_\_\_\_\_\_\_\_Щека С. А.\_\_\_\_\_\_\_\_\_\_

(Фамилия И.О.)

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Проверил: \_Асс. Кононенко К.М.*\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_***

(Фамилия И.О., степень, звание) **(Оценка)**

**Дата, подпись** \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ***\_\_\_\_\_\_\_\_\_\_\_***

(Дата) (Подпись)

**Замечания: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Москва**

**2019**

Оглавление

[**Теория** 3](#_Toc20673720)

[**Задания** 4](#_Toc20673721)

[**Блок-схемы** 5](#_Toc20673722)

[**Коды программ** 7](#_Toc20673723)

[Листинг 1 7](#_Toc20673724)

[Листинг 2 8](#_Toc20673725)

[Листинг 3 9](#_Toc20673726)

[Листинг 4 10](#_Toc20673727)

[Листинг 5 11](#_Toc20673728)

[**Результаты выполнения програм** 12](#_Toc20673729)

# **Теория**

При вызове функции ей при помощи аргументов (формальных параметров) могут быть переданы некоторые значения (фактические параметры), используемые во время выполнения функции. Функция может возвращать некоторое (одно !) значение. Это возвращаемое значение и есть результат выполнения функции, который при выполнении программы подставляется в точку вызова функции, где бы этот вызов ни встретился. Допускается также использовать функции не имеющие аргументов и функции не возвращающие никаких значений. Действие таких функций может состоять, например, в изменении значений некоторых переменных, выводе на печать некоторых текстов и т.п..

С использованием функций в языке СИ связаны три понятия - определение функции (описание действий, выполняемых функцией), объявление функции (задание формы обращения к функции) и вызов функции.

Определение функции задает тип возвращаемого значения, имя функции, типы и число формальных параметров, а также объявления переменных и операторы, называемые телом функции, и определяющие действие функции. В определении функции также может быть задан класс памяти.

В программах на языке СИ широко используются, так называемые, библиотечные функции, т.е. функции предварительно разработанные и записанные в библиотеки. Прототипы библиотечных функций находятся в специальных заголовочных файлах, поставляемых вместе с библиотеками в составе систем программирования, и включаются в программу с помощью директивы #include.

Если объявление функции не задано, то по умолчанию строится прототип функции на основе анализа первой ссылки на функцию, будь то вызов функции или определение. Однако такой прототип не всегда согласуется с последующим определением или вызовом функции. Рекомендуется всегда задавать прототип функции. Это позволит компилятору либо выдавать диагностические сообщения, при неправильном использовании функции, либо корректным образом регулировать несоответствие аргументов устанавливаемое при выполнении программы.

# **Задания**

1. Описать функцию PowerA3(A, B), вычисляющую третью степень числа A и возвращающую ее в переменной B (A — входной, B — выходной параметр; оба параметра являются вещественными). С помощью этой функции найти третьи степени пяти данных чисел.

2. Описать функцию Sign(X) целого типа, возвращающую для вещественного числа X следующие значения:

−1, если X < 0; 0, если X = 0; 1, если X > 0.

С помощью этой функции найти значение выражения Sign(A) + Sign(B) для данных вещественных чисел A и B.

3. Описать функцию RingS(R1, R2) вещественного типа, находящую площадь кольца, заключенного между двумя окружностями с общим центром и радиусами R1 и R2 (R1 и R2 — вещественные, R1 > R2). С ее помощью найти площади трех колец, для которых даны внешние и внутренние радиусы.

4. Описать функцию Quarter(x, y) целого типа, определяющую номер координатной четверти, в которой находится точка с ненулевыми вещественными координатами (x, y). С помощью этой функции найти номера координатных четвертей для трех точек с данными ненулевыми координатами

5. Описать функцию Fact2(N) вещественного типа, вычисляющую двойной факториал:

N!! = 1·3·5·. . .·N, если N — нечетное;

N!! = 2·4·6·. . .·N, если N — четное (N > 0 — параметр целого типа; вещественное возвращаемое значение используется для того, чтобы избежать целочисленного переполнения при больших значениях N).

# **Блок-схемы**

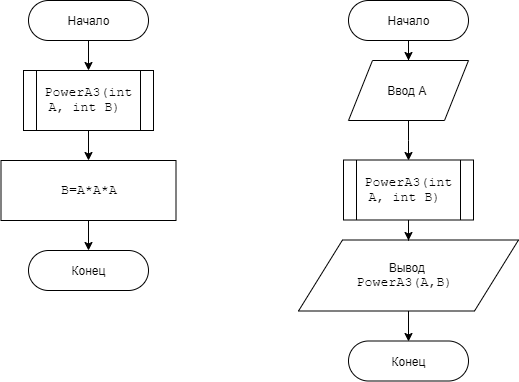


Рисунок 1 — Блок-схема к заданию 1.

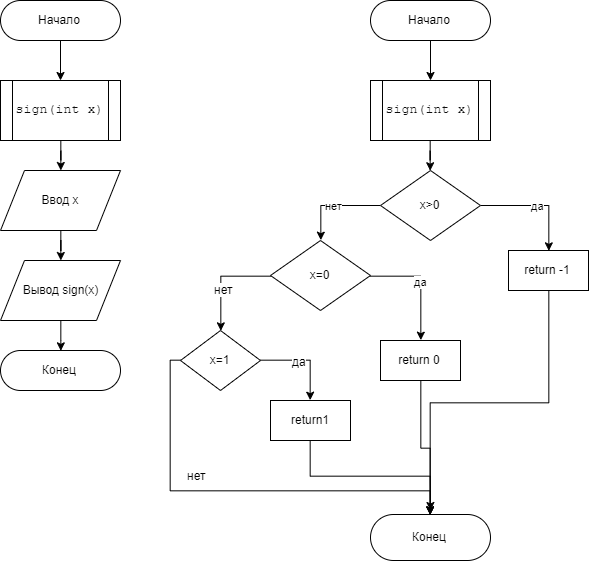


Рисунок 2 — Блок-схема к заданию 2.

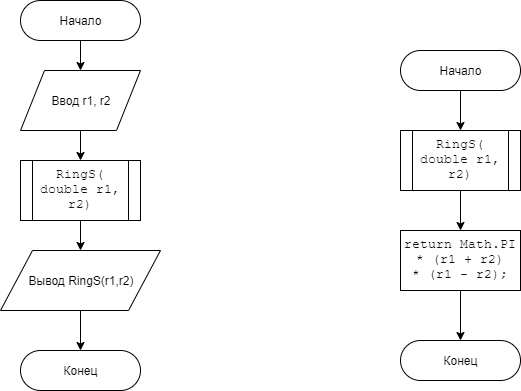


Рисунок 3 — Блок-схема к заданию 3.

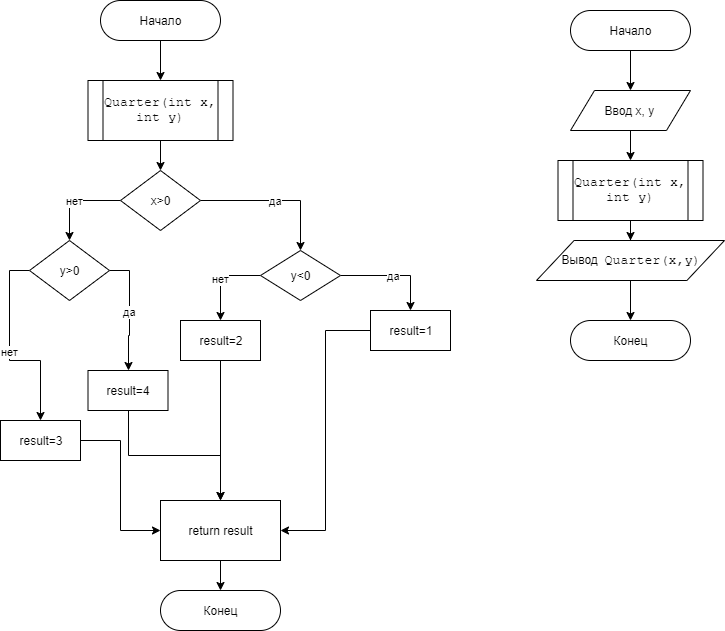


Рисунок 4 — Блок-схема к заданию 4.

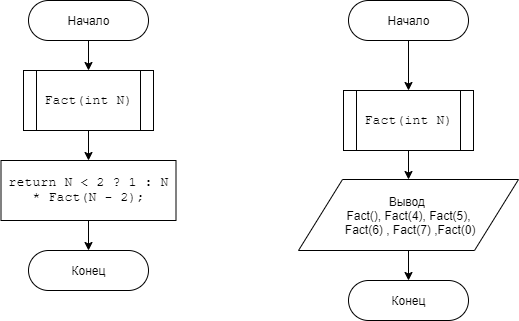


Рисунок 5 — Блок-схема к заданию 5.

# **Коды программ**

Листинг 1 —Задание 1(Третья степень числа)

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace ConsoleApp1
7. {
8. class Program
9. {
10. static public int PowerA3(int A, int B )
11. {
12. return B = A \* A \* A;
13. }
14. static void Main(string[] args)
15. {
16. int B=0;
17. Console.Write(" А = ");
18. int A = Convert.ToInt32(Console.ReadLine());
20. Console.WriteLine("" + PowerA3(A,B));
21. Console.ReadLine();
22. }
23. }
24. }

Листинг 2 —Задание 2 (Нахождение суммы)

1. using System.Threading.Tasks;
2. using System;
3. using System.Collections.Generic;
4. using System.Linq;
5. using System.Text;
6. using System.Threading.Tasks;
7. namespace ConsoleApp1
8. {
10. class Program
11. {
12. static public int sign(int x)
13. {
14. if (x > 0)
15. return x = -1;
16. else if (x == 0) return x = 0;
17. else return x = 1;
18. }
19. static void Main(string[] args)
20. {
22. Console.Write(" a = ");
23. int a = Convert.ToInt32(Console.ReadLine());
24. Console.Write(" b = ");
25. int b = Convert.ToInt32(Console.ReadLine());
27. Console.WriteLine(" " + sign(a)+sign(b) );
28. Console.ReadLine();
29. }
31. }
32. }

Листинг 3 —Задание 3 (Функция радиуса)

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace ConsoleApp1
7. {
8. class Program
9. {
10. static double RingS(double r1, double r2)
11. {
12. return Math.PI \* (r1 + r2) \* (r1 - r2);
13. }
14. static void Main(string[] args)
15. {
16. Console.Write(" R1 = ");
17. double r1 = Convert.ToDouble(Console.ReadLine());
18. Console.Write(" R2 = ");
19. double r2 = Convert.ToDouble(Console.ReadLine());
20. Console.WriteLine(" " + RingS(r1, r2) );
21. Console.ReadLine();
22. }
23. }
24. }

Листинг 4 —Задание 4 (Нахождение четверти)

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace ConsoleApp1
7. {
8. class Program
9. {
10. public static int Quarter(int x, int y)
11. {
12. int result = 0;
13. if (x > 0)
14. {
15. if (y > 0)
16. result = 1;
17. else if (y < 0)
18. result = 2;
19. }
20. else
21. {
22. if (y > 0)
23. result = 4;
24. else if (y < 0)
25. result = 3;
26. }
27. return result;
28. }
30. static void Main(string[] args)
31. {
32. Console.Write(" x = ");
33. int x = Convert.ToInt32(Console.ReadLine());
34. Console.Write(" y = ");
35. int y = Convert.ToInt32(Console.ReadLine());
36. Console.WriteLine(" четверть " + Quarter(x,y));
38. Console.ReadLine();
39. }
40. }
41. }

Листинг 5 - Задание 5 (Нахождение двойного факториала )

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace ConsoleApp5
7. {
8. class Program
9. {
10. public static double Fact(int N)
11. { int i, s=1;
13. if ((N % 2) == 0)
14. {
15. for (i = 2; i < N; i = i + 2)
16. {
17. s = i \* s;
18. }
19. } else
20. {
21. for (i = 1; i < s; i = i + 2)
22. {
23. s = i \* s;
24. }
25. }
26. return s;
28. }
29. static void Main(string[] args)
30. {
31. Console.Write(" N = ");
32. int N = Convert.ToInt32(Console.ReadLine());
33. Console.WriteLine(Fact(N));
34. Console.ReadKey();
36. }
37. }
38. }

# **Результаты выполнения программ**



Рисунок 6 — результат выполнения программы 1.

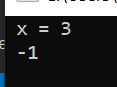


Рисунок 7 — результат выполнения программы 2.

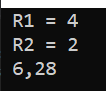


Рисунок 8 — результат выполнения программы 3.

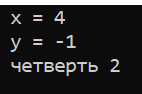


Рисунок 9 — результат выполнения программы 4.

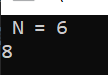


Рисунок 10 — результат выполнения программы 5.