**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 № 1**

**Дисциплина:** Введение в программирование.

**Выполнил(а):**

**студент(ка) группы 191-726**

Филатов А.К.

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

(Дата) (Подпись)

**Проверил:** асс. Кононенко К.М.

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

(Дата) (Подпись)

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

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

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

**Москва**

**2019**

Оглавление

[**Теория** 2](#_Toc20500756)

[**Задания** 4](#_Toc20500757)

[**Блок-схемы** 5](#_Toc20500758)

[**Код программы** 9](#_Toc20500759)

[**Результат программы** 14](#_Toc20500760)

# 

# **Теория**

В следующей таблице перечислены операторы C# в порядке убывания приоритета. Операторы в каждой строке имеют одинаковый приоритет.

x.y, x?.y, x?[y], f(x), a[i], x++, x--, new, typeof, checked, unchecked, default, nameof, delegate, sizeof, stackalloc, x->y - Первичный

+x, -x, !x, ~x, ++x, --x, ^x, (T)x, await, &x, \*x, true и false - Унарный

x..y - Диапазон

x \* y, x / y, x % y - Мультипликативный

x + y, x – y - Аддитивный

x << y, x >> y - Сдвиг

x < y, x > y, x <= y, x >= y, is, as - Тестирование типов и относительный

x == y, x != y - Равенство

x & y - Логическое И или побитовое логическое И

x ^ y - Логическое исключающее ИЛИ или побитовое логическое исключающее ИЛИ

x | y - Логическое ИЛИ или побитовое логическое ИЛИ

x && y - Условное И

x || y - Условное ИЛИ

x ?? y - Оператор объединения с NULL

c ? t : f - Условный оператор

x = y, x += y, x -= y, x \*= y, x /= y, x %= y, x &= y, x |= y, x ^= y, x <<= y, x >>= y, x ??= y, => - Назначение и объявление лямбда-выражений

# **Задания**

1. Даны стороны прямоугольника a и b. Найти его площадь S = a·b и  
периметр P = 2·(a + b)

2. Дан диаметр окружности d. Найти ее длину L = π·d. В качестве  
значения π использовать 3.14.

3. Даны два числа a и b. Найти их среднее арифметическое: (a + b)/2.

4. Даны два ненулевых числа. Найти сумму, разность, произведение и  
частное их квадратов.

5. Даны два ненулевых числа. Найти сумму, разность, произведение и  
частное их модулей.

**Блок-схемы**

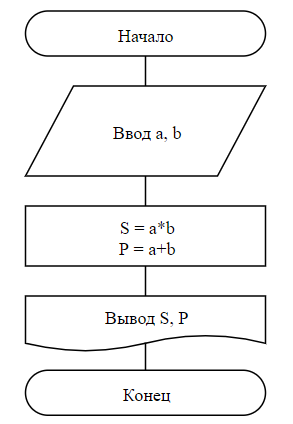


Рисунок 1 — Блок-схема к заданию 1

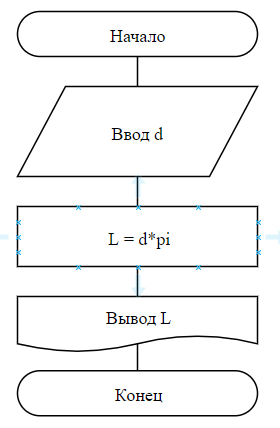


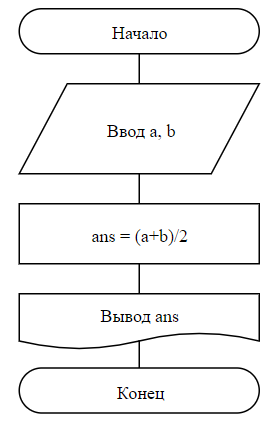
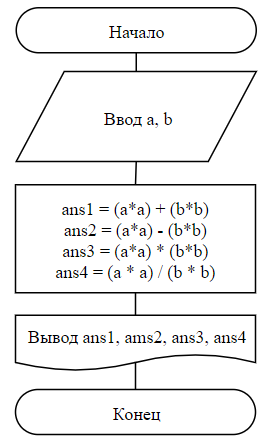
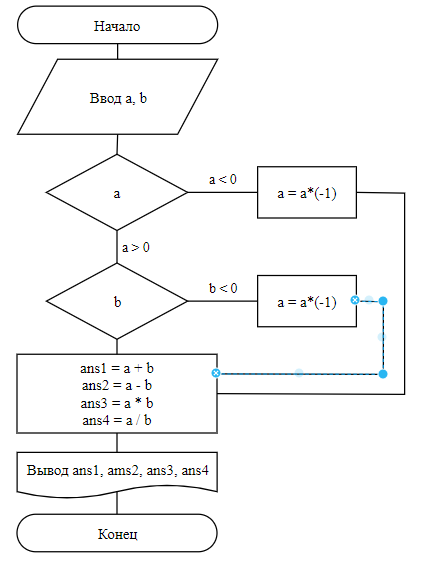
Рисунок 2 — Блок-схема к заданию 2   
  
   
  
Рисунок 3 — Блок-схема к заданию 3   
 

Рисунок 4 — Блок-схема к заданию 4   
  
   
  
Рисунок 5 — Блок-схема к заданию 5

# 

# **Код программы**

Листинг 1 — Задание 1

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. namespace LaboratorySolution 6. { 7. class Program 8. { 9. static void Main(string[] args) 10. { 11. int a, b; // создание переменных 12. a = int.Parse(Console.ReadLine()); // ввод переменных 13. b = int.Parse(Console.ReadLine()); 14. Console.WriteLine("S = " + a \* b); // вывод ответа 15. Console.WriteLine("P = " + 2 \* (a + b)); 16. Console.ReadLine(); // пауза для просмотра результата 17. } 18. } 19. } |

Листинг 2 — Задание 2

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. namespace LaboratorySolution 6. { 7. class Program 8. { 9. static void Main(string[] args) 10. { 11. double d, pi = 3.14; // создание переменных 12. d = int.Parse(Console.ReadLine()); // ввод переменных 13. Console.WriteLine("L = " + d\*pi); // вывод ответа 14. Console.ReadLine(); // пауза для просмотра результата 15. } 16. } 17. } |

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. namespace LaboratorySolution 6. { 7. class Program 8. { 9. static void Main(string[] args) 10. { 11. double a, b; // создание переменных 12. a = int.Parse(Console.ReadLine()); // ввод переменных 13. b = int.Parse(Console.ReadLine()); 14. Console.WriteLine("(a+b)/2 = " + (a+b)/2); // вывод ответа 15. Console.ReadLine(); // пауза для просмотра результата 16. } 17. } 18. } |

Листинг 3 — Задание 3

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. namespace LaboratorySolution 6. { 7. class Program 8. { 9. static void Main(string[] args) 10. { 11. int a, b; // создание переменных 12. a = int.Parse(Console.ReadLine()); // ввод переменных 13. b = int.Parse(Console.ReadLine()); 14. Console.WriteLine("a^2 + b^2 = " + (a\*a) + (b\*b)); // вывод ответа 15. Console.WriteLine("a^2 - b^2 = " + ((a \* a) - (b \* b)) ); 16. Console.WriteLine("a^2 \* b^2 = " + ((a \* a) \* (b \* b)) ); 17. Console.WriteLine("a^2 / b^2 = " + ((a \* a) / (b \* b)) ); 18. Console.ReadLine(); // пауза для просмотра результата 19. } 20. } 21. } |

Листинг 4 — Задание 4 (Нахождение периметра и площади прямоугольника)

Листинг 5 — Задание 5 (Нахождение периметра и площади треугольника)

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. namespace LaboratorySolution 6. { 7. class Program 8. { 9. static void Main(string[] args) 10. { 11. int a, b; // создание переменных 12. a = int.Parse(Console.ReadLine()); // ввод переменных 13. b = int.Parse(Console.ReadLine()); 14. if (a < 0) a = a \* (-1); 15. if (b < 0) b = b \* (-1); 16. Console.WriteLine("a + b = " + (a + b)); // вывод ответа 17. Console.WriteLine("a - b = " + (a - b)); 18. Console.WriteLine("a \* b = " + (a \* b)); 19. Console.WriteLine("a / b = " + (a / b)); 20. Console.ReadLine(); // пауза для просмотра результата 21. } 22. } 23. } |

# 

# **Результат программы**

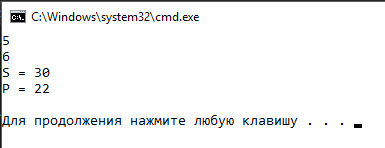


Рисунок 6 — Результат выполнения программы 1

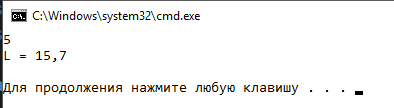


Рисунок 7 — Результат выполнения программы 2

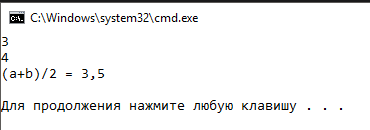


Рисунок 8 — Результат выполнения программы 3

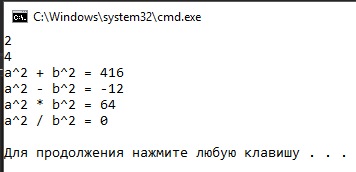


Рисунок 9 — Результат выполнения программы 4

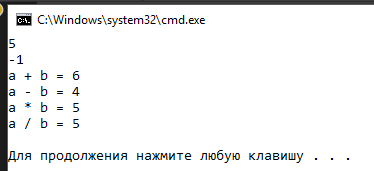


Рисунок 10 — Результат выполнения программы 5