**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 № 2**

**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**

Логические операторы, которые сравнивают две логические величины:

- логическое И - оператор && - возвращает истину только в том случае, если и справа и слева от него будет истина, во всех остальных случаях будет ложь

- логическое ИЛИ - оператор || - возвращает истину, если хоть одна из двух величин истинна. Ложь он вернет, только если обе логические величины ложны

В языке C# предоставляется целый класс математических методов. Это класс - Math. В этом классе методы статические.

Таблица 1 – Таблица математических методов.

|  |  |  |
| --- | --- | --- |
| Math.Abs | Возвращаем абсолютное число, имеет 7 перегрузок. То есть метод принимает разные типы переменных. | int i = Math.Abs(x); |
| Math.Acos | Арк Косинус. Определяется угол, косинус которого равен указанному числу. | double i = Math.Acos(0.5); |
| Math.Asin | Арк Синус. Также определяет угол. | double i = Math.Asin(0.5); |
| Math.Atan | Арк Тангенс. Возвращает угол, значение которого было указано | double i = Math.Atan(0.5); |
| Math.Cos | Возвращает косинус угла. | double x = Math.Cos(1.04); |
| Math.Cosh | Возвращает гиперболический косинус угла. | double x = Math.Cosh(radian); |
| Math.Exp | Экспонента. | double x = Math.Exp(2); |
| Math.Log | Вычисление логарифма. X - число которое нужно найти, Osn - основание логарифма. | double x = Math.Log(X,Osn); |
| Math.Log10 | Вычисление десятичного логарифма. | double x = Math.Log10(10) |
| Math.Max | Возвращает из 2-х чисел большее число. Имеет 11 перегруженых методов. | int x = Math.Max(10,20); |
| Math.Min | Возвращает из 2-х чисел меньшее число. Имеет 11 перегруженых методов. | int x = Math.Min(10,20); |
| Math.PI | Возвращает число Пи. | double pi = Math.PI; |
| Math.Pow | Вычисляет число возведенное в степень: ax | double i = Math.Pow(a, x); |
| Math.Sin | Возвращает синус угла. | double p = Math.Sin(0.5); |
| Math.Sinh | Возвращает гиперболический синус угла. | double p = Math.Sin(0.5); |
| Math.Sqrt | Возвращает квадратный корень. | double r = Math.Sqrt(7); |
| Math.Tan | Возвращает тангенс угла. | double p = Math.Tan(1.04); |
| Math.Tanh | Возвращает гиперболический тангенс угла. | double p = Math.Tanh(1.04); |

# 

# **Задания**

1. Найти расстояние между двумя точками с заданными координатами (x1, y1) и (x2, y2)

2. Даны три точки A, B, C на числовой оси. Найти длины отрезков AC

и BC и их сумму.

3. Даны три точки A, B, C на числовой оси. Точка C расположена

между точками A и B. Найти произведение длин отрезков AC и BC

4. Даны координаты двух противоположных вершин прямоугольника:

(x1, y1), (x2, y2). Стороны прямоугольника параллельны осям координат.

Найти периметр и площадь данного прямоугольника.

5. Даны координаты трех вершин треугольника: (x1, y1), (x2, y2), (x3, y3). Найти его периметр и площадь

**Блок-схемы**

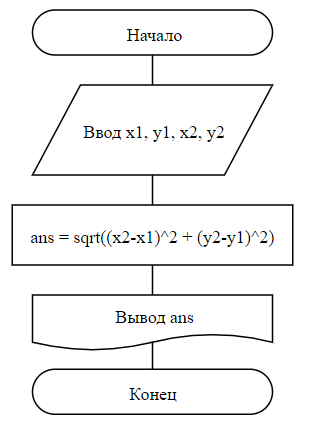


Рисунок 1 — Блок-схема к заданию 1

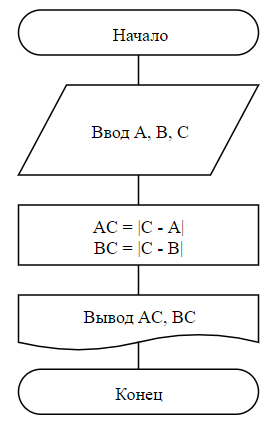
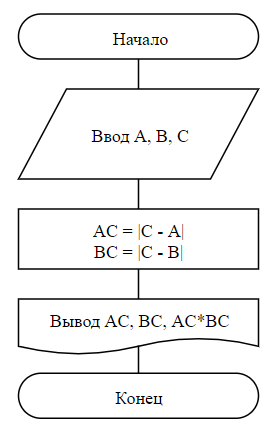


Рисунок 2 — Блок-схема к заданию 2   
  
   
  
Рисунок 3 — Блок-схема к заданию 3

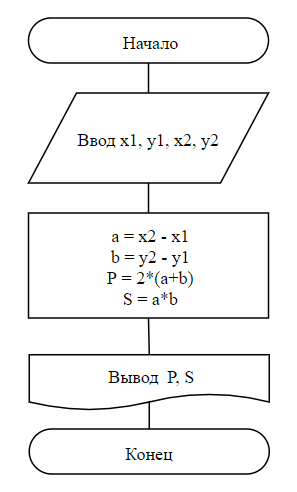
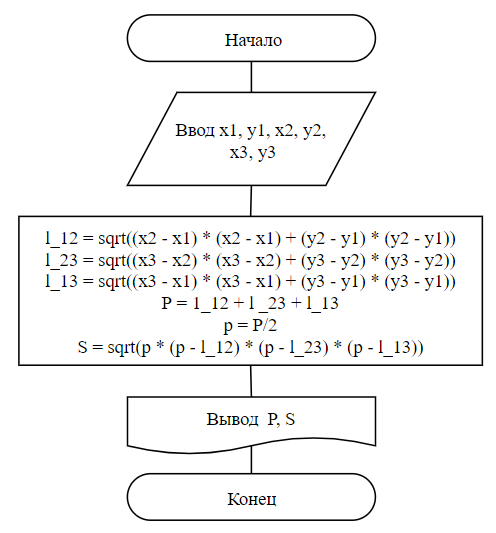


Рисунок 4 — Блок-схема к заданию 4   
  
   
  
Рисунок 5 — Блок-схема к заданию 5

# **Код программы**

Листинг 1 — Задание 1(Нахождение расстояния между двумя точками с заданными координатами)

|  |
| --- |
| 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 x1, x2, y1, y2, ans, inter1, inter2; // создание переменных 12. x1 = double.Parse(Console.ReadLine()); // ввод переменных 13. y1 = double.Parse(Console.ReadLine()); 14. x2 = double.Parse(Console.ReadLine()); 15. y2 = double.Parse(Console.ReadLine()); 16. inter1 = x2 - x1; 17. inter2 = y2 - y1; 18. ans = Math.Sqrt(inter1 \* inter1 + inter2 \* inter2); 19. Console.WriteLine("Length: " + ans); 20. Console.ReadLine(); // пауза для просмотра результата 21. } 22. } 23. } |

Листинг 2 — Задание 2 (Нахождение длины отрезков и их сумму)

|  |
| --- |
| 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, c, ac, bc; // создание переменных 12. a = double.Parse(Console.ReadLine()); // ввод переменных 13. b = double.Parse(Console.ReadLine()); 14. c = double.Parse(Console.ReadLine()); 15. ac = Math.Abs(c - a); 16. bc = Math.Abs(c - b); 17. Console.WriteLine("AC length = " + ac); 18. Console.WriteLine("BC length = " + bc); 19. Console.WriteLine("AC + BC length = " + (ac + bc)); 20. Console.ReadLine(); // пауза для просмотра результата 21. } 22. } 23. } |

|  |
| --- |
| 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, c, ac, bc; // создание переменных 12. a = double.Parse(Console.ReadLine()); // ввод переменных 13. b = double.Parse(Console.ReadLine()); 14. c = double.Parse(Console.ReadLine()); 15. ac = Math.Abs(c – a); 16. bc = Math.Abs(b – c); 17. Console.WriteLine("AC length = " + ac); 18. Console.WriteLine("BC length = " + bc); 19. Console.WriteLine("AC \* BC length = " + (ac \* bc)); 20. Console.ReadLine(); // пауза для просмотра результата 21. } 22. } 23. } |

Листинг 3 — Задание 3 (Нахождение произведения длин отрезков)

|  |
| --- |
| 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 x1, y1, x2, y2, a, b; // создание переменных 12. Console.WriteLine("Input x1, y1, x2, y2: "); 13. x1 = double.Parse(Console.ReadLine()); // ввод переменных 14. y1 = double.Parse(Console.ReadLine()); 15. x2 = double.Parse(Console.ReadLine()); 16. y2 = double.Parse(Console.ReadLine()); 17. a = Math.Abs(x2 - x1); // вычисление сторон прямоугольника 18. b = Math.Abs(y2 - y1); 19. Console.WriteLine("P = " + 2 \* (a + b)); 20. Console.WriteLine("S = " + (a \* b)); 21. Console.ReadLine(); // пауза для просмотра результата 22. } 23. } 24. } |

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. double x1, y1, x2, y2, x3, y3, l\_12, l\_23, l\_13, p, hp, s; // создание переменных 12. Console.WriteLine("Input x1, y1, x2, y2, x3, y3: "); 13. x1 = double.Parse(Console.ReadLine()); // ввод переменных 14. y1 = double.Parse(Console.ReadLine()); 15. x2 = double.Parse(Console.ReadLine()); 16. y2 = double.Parse(Console.ReadLine()); 17. x3 = double.Parse(Console.ReadLine()); 18. y3 = double.Parse(Console.ReadLine()); 19. l\_12 = Math.Sqrt((x2 - x1) \* (x2 - x1) + (y2 - y1) \* (y2 - y1)); 20. l\_23 = Math.Sqrt((x3 - x2) \* (x3 - x2) + (y3 - y2) \* (y3 - y2)); 21. l\_13 = Math.Sqrt((x3 - x1) \* (x3 - x1) + (y3 - y1) \* (y3 - y1)); 22. p = l\_12 + l\_13 + l\_23; 23. hp = (l\_23 + l\_13 + l\_12) / 2; 24. s = Math.Sqrt(hp \* (hp - l\_12) \* (hp - l\_23) \* (hp - l\_13)); 25. Console.WriteLine("P = " + p); 26. Console.WriteLine("S = " + s); 27. Console.ReadLine(); // пауза для просмотра результата 28. } 29. } 30. } |

# 

# **Результат программы**

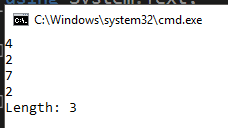


Рисунок 6 — Результат выполнения программы 1

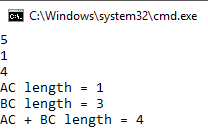


Рисунок 7 — Результат выполнения программы 2

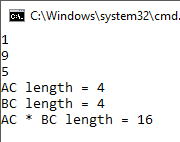


Рисунок 8 — Результат выполнения программы 3

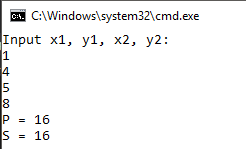


Рисунок 9 — Результат выполнения программы 4

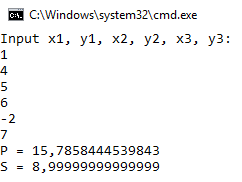


Рисунок 10 — Результат выполнения программы 5