**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**

Синельникова К.Т.

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

(Дата) (Подпись)

**Проверил:** асс. Кононенко К.М.

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

(Дата) (Подпись)

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

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

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

**Москва**

**2019**

Оглавление

[**Теория** 3](#_Toc20500756)

[**Задания** 5](#_Toc20500757)

[**Блок-схемы** 6](#_Toc20500758)

[**Код программы** 9](#_Toc20500759)

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

# 

# **Теория**

Сделать поведение программы еще более сложным нам помогут специальные логические операторы, которые сравнивают две логические величины (каждая из которых может быть либо истиной, либо ложью)

- 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.

|  |  |  |
| --- | --- | --- |
| 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. using System.Threading.Tasks; 6. namespace Код\_Лабораторной\_2\_\_1\_5\_\_шарп 7. { 8. class Program 9. { 10. static void Main(string[] args) 11. { 12. float x1, x2, y1, y2, a, b; 13. Console.WriteLine("Введите координаты точек: "); 14. Console.Write("x1 = "); 15. x1 = float.Parse(Console.ReadLine()); 16. Console.Write("y1 = "); 17. y1 = float.Parse(Console.ReadLine()); 18. Console.Write("x2 = "); 19. x2 = float.Parse(Console.ReadLine()); 20. Console.Write("y2 = "); 21. y2 = float.Parse(Console.ReadLine()); 22. Console.WriteLine("Координаты точки 1: (" + x1 + "; " + y1 + ")"); 23. Console.WriteLine("Координаты точки 2: (" + x2 + "; " + y2 + ")"); 24. double d = Math.Sqrt(Math.Abs((x2 - x1)\* (x2 - x1) + (y2 - y1)\*(y2 - y1))); 25. Console.WriteLine("Расстояние между точками: " + d); 26. Console.ReadKey(); 27. } 28. } 29. } |

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. using System.Threading.Tasks; 6. namespace Код\_Лабораторной\_2\_\_1\_5\_\_шарп 7. { 8. class Program 9. { 10. static void Main(string[] args) 11. { 12. float A, B, C; 13. Console.WriteLine("Введите значения трёх точек: "); 14. Console.Write("A = "); 15. A = float.Parse(Console.ReadLine()); 16. Console.Write("B = "); 17. B = float.Parse(Console.ReadLine()); 18. Console.Write("C = "); 19. C = float.Parse(Console.ReadLine()); 20. float AC = Math.Abs(C - A); 21. Console.WriteLine("Длина отрезка AC = " + AC); 22. float BC = Math.Abs(C - B); 23. Console.WriteLine("Длина отрезка BC = " + BC); 24. Console.WriteLine("Сумма отрезков AC и BC = " + (AC+BC)); 25. Console.ReadKey(); 26. } 27. } 28. } |

|  |
| --- |
| 1. using System; 2. using System.Collections.Generic; 3. using System.Linq; 4. using System.Text; 5. using System.Threading.Tasks; 6. namespace Код\_Лабораторной\_2\_\_1\_5\_\_шарп 7. { 8. class Program 9. { 10. static void Main(string[] args) 11. { 12. float A, B, C; 13. Console.WriteLine("Обязательное условие задачи - точка C лежит между точками A и B"); 14. Console.WriteLine("Введите значения трёх точек: "); 15. Console.Write("A = "); 16. A = float.Parse(Console.ReadLine()); 17. Console.Write("B = "); 18. B = float.Parse(Console.ReadLine()); 19. Console.Write("C = "); 20. C = float.Parse(Console.ReadLine()); 21. if (C > A && C < B || C > B && C < A) 22. { 23. Console.WriteLine("Произведение АС и ВС: " + ((Math.Abs(C - A)) \* (Math.Abs(B - C)))); 24. } 25. else 26. { 27. Console.WriteLine("Ошибка. Введите значения заново."); 28. } 29. Console.ReadKey();} 30. } 31. } 32. } |

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. using System.Threading.Tasks; 6. namespace Код\_Лабораторной\_2\_\_1\_5\_\_шарп 7. { 8. class Program 9. { 10. static void Main(string[] args) 11. { 12. int x1, x2, y1, y2; 13. Console.WriteLine("Введите координаты точки 1: "); 14. Console.Write("x1 = "); 15. x1 = int.Parse(Console.ReadLine()); 16. Console.Write("y1 = "); 17. y1 = int.Parse(Console.ReadLine()); 18. Console.WriteLine("Введите координаты точки 2: "); 19. Console.Write("x2 = "); 20. x2 = int.Parse(Console.ReadLine()); 21. Console.Write("y2 = "); 22. y2 = int.Parse(Console.ReadLine()); 23. Console.WriteLine("Периметр равен " + (2 \* (Math.Abs(x2 - x1) + Math.Abs(y2 - y1)))); 24. Console.WriteLine("Площадь равна " + (Math.Abs(x2 - x1) \* Math.Abs(y2 - y1))); 25. Console.ReadKey(); 26. } 27. } 28. } |

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. using System.Threading.Tasks; 6. namespace Код\_Лабораторной\_2\_\_1\_5\_\_шарп 7. { 8. class Program 9. { 10. static void Main(string[] args) 11. { 12. float x1, x2, x3, y1, y2, y3; 13. Console.WriteLine("Введите координаты вершины 1: "); 14. Console.Write("x1 = "); 15. x1 = float.Parse(Console.ReadLine()); 16. Console.Write("y1 = "); 17. y1 = float.Parse(Console.ReadLine()); 18. Console.WriteLine("Введите координаты вершины 2: "); 19. Console.Write("x2 = "); 20. x2 = float.Parse(Console.ReadLine()); 21. Console.Write("y2 = "); 22. y2 = float.Parse(Console.ReadLine()); 23. Console.WriteLine("Введите координаты вершины 3: "); 24. Console.Write("x3 = "); 25. x3 = float.Parse(Console.ReadLine()); 26. Console.Write("y3 = "); 27. y3 = float.Parse(Console.ReadLine()); 28. Console.WriteLine("Площадь равна " + (0.5 \* Math.Abs((x2 - x1) \* (y3 - y1) - (x3 - x1) \* (y2 - y1)))); 29. Console.WriteLine("Периметр равен " + (Math.Sqrt((x2 - x1) \* (x2 - x1) + (y2 - y1) \* (y2 - y1)) + Math.Sqrt((x3 - x2) \* (x3 - x2) + (y3 - y2) \* (y3 - y2)) + Math.Sqrt((x3 - x1) \* (x3 - x1) + (y3 - y1) \* (y3 - y1)))); 30. Console.ReadKey(); 31. } 32. } 33. } |

# 

# **Результат программы**

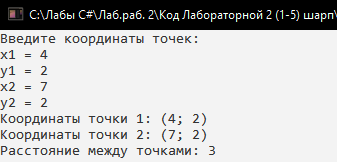


Рисунок 6 — Результат выполнения программы 1

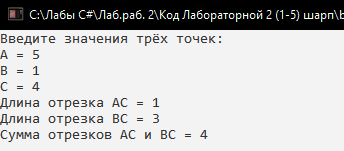


Рисунок 7 — Результат выполнения программы 2

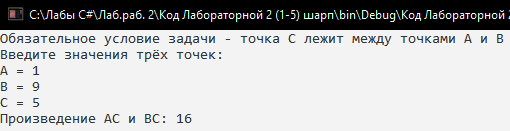


Рисунок 8 — Результат выполнения программы 3

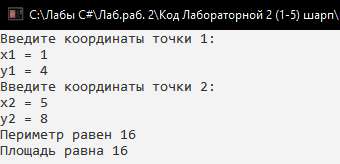


Рисунок 9 — Результат выполнения программы 4

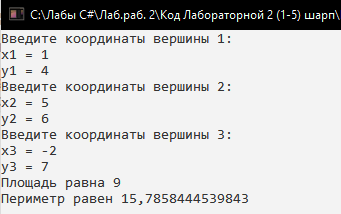


Рисунок 10 — Результат выполнения программы 5