**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\_**

\_\_\_\_\_\_\_\_Савеленко В.Н.\_\_\_\_\_\_\_\_\_\_\_

(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](#_Toc19546885)

[**Задания** 4](#_Toc19546886)

[**Блок-схемы** 5](#_Toc19546887)

[**Коды программ** 8](#_Toc19546888)

[Листинг 1 8](#_Toc19546889)

[Листинг 2 8](#_Toc19546890)

[Листинг 3 9](#_Toc19546891)

[Листинг 4 10](#_Toc19546892)

[Листинг 5 11](#_Toc19546893)

[**Результаты выполнения программ** 12](#_Toc19546894)

# **Теория**

Целочисленный тип данных в [информатике](https://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%84%D0%BE%D1%80%D0%BC%D0%B0%D1%82%D0%B8%D0%BA%D0%B0) — один из простейших и распространённых [типов данных](https://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) в [языках программирования](https://ru.wikipedia.org/wiki/%D0%AF%D0%B7%D1%8B%D0%BA_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F). Служит для представления [целых чисел](https://ru.wikipedia.org/wiki/%D0%A6%D0%B5%D0%BB%D0%BE%D0%B5_%D1%87%D0%B8%D1%81%D0%BB%D0%BE).

Множество чисел этого типа представляет собой конечное [подмножество](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%B4%D0%BC%D0%BD%D0%BE%D0%B6%D0%B5%D1%81%D1%82%D0%B2%D0%BE) бесконечного множества целых чисел, ограниченное [максимальным](https://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D0%BA%D1%81%D0%B8%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D1%8D%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82) и [минимальным](https://ru.wikipedia.org/wiki/%D0%9C%D0%B8%D0%BD%D0%B8%D0%BC%D0%B0%D0%BB%D1%8C%D0%BD%D1%8B%D0%B9_%D1%8D%D0%BB%D0%B5%D0%BC%D0%B5%D0%BD%D1%82) значениями.

Константы — это постоянные значения, которые известны во время компиляции и не изменяются во время выполнения программы. Константы должны объявляться с модификатором [const](https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/const).

Символы // преобразуют остальную часть строки в комментарий.

Консольное приложение C# должно содержать метод Main, в котором начинается и заканчивается управление. В методе Main создаются объекты и выполняются другие методы.

Метод Main является [статическим](https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/static) методом, расположенным внутри класса или структуры.

Программы на C#, как правило, используют службы ввода-вывода, предоставляемые библиотекой времени выполнения в .NET Framework. Инструкция System.Console.WriteLine(); использует метод [WriteLine](https://docs.microsoft.com/ru-ru/dotnet/api/system.console.writeline). Это один из методов вывода класса [Console](https://docs.microsoft.com/ru-ru/dotnet/api/system.console) в библиотеке времени выполнения. Он отображает свой строковый параметр в стандартном потоке вывода, за которым следует новая строка. Существуют и другие методы [Console](https://docs.microsoft.com/ru-ru/dotnet/api/system.console) для разных операций ввода и вывода. Если вы добавите в начало программы директиву using System;, классы и методы [System](https://docs.microsoft.com/ru-ru/dotnet/api/system) можно использовать напрямую, не указывая их полные имена. Например, можно вызвать Console.WriteLine вместо System.Console.WriteLine.

# **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

# **Коды программ**

Листинг 1 — Задание 1(Нахождение расстояния между точками)

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Laboratory\_2
7. {
8. class Program
9. {
10. static void Main(string[] args)
11. {
12. int x1, x2, y1, y2;
13. Console.WriteLine("Введите координаты 1 точки (x1;y1) по-очередно: ");
14. x1 = int.Parse(Console.ReadLine());
15. y1 = int.Parse(Console.ReadLine());
16. Console.WriteLine("Введите y координаты 2 точки (x2;y2) по-очередно: ");
17. x2 = int.Parse(Console.ReadLine());
18. y2 = int.Parse(Console.ReadLine());
19. Console.WriteLine("Расстояние между этими точками = " + (Math.Sqrt((Math.Pow((y1 - x1), 2) + Math.Pow((y2-x2),2)))));
20. Console.ReadLine();
21. }}}

Листинг 2 — Задание 2 (Нахождение 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 Laboratory\_2
7. {
8. class Program
9. {
10. static void Main(string[] args)
11. {
12. int A, B, C;
13. Console.WriteLine("Введите значения точек A, B и С по-очередно: ");
14. A = int.Parse(Console.ReadLine());
15. B = int.Parse(Console.ReadLine());
16. C = int.Parse(Console.ReadLine());
17. Console.WriteLine("Длина отрезка АС = " + Math.Abs(C - A));
18. Console.WriteLine("Длина отрезка BС = " + Math.Abs(C - B));
19. Console.WriteLine("Сумма отрезков AC и BC = " + Math.Abs((C - A) + (C - B)));
20. Console.ReadLine();
21. }}}

Листинг 3 — Задание 3 (Произведение 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 Laboratory\_2
7. {
8. class Program
9. {
10. static void Main(string[] args)
11. {
12. int A, B, C;
13. Console.WriteLine("Введите точки А, В и С так, что бы С было между А и В: ");
14. A = int.Parse(Console.ReadLine());
15. B = int.Parse(Console.ReadLine());
16. C = int.Parse(Console.ReadLine());
17. Console.WriteLine("Произведение длин отрезков AC и BC = " + ((C - A) \* (B - C)));
18. Console.ReadLine();
19. }}}

Листинг 4 — Задание 4 (Вычисление периметра и площади прямоугольника по координатам точек его вершин)

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Laboratory\_2
7. {
8. class Program
9. {
10. static void Main(string[] args)
11. {
12. int x1, x2, y1, y2;
13. Console.WriteLine("Введите координаты противоположных точек прямоугольника: ");
14. Console.WriteLine("Введите координаты 1 точки (x1;y1) по-очередно: ");
15. x1 = int.Parse(Console.ReadLine());
16. y1 = int.Parse(Console.ReadLine());
17. Console.WriteLine("Введите y координаты 2 точки (x2;y2) по-очередно: ");
18. x2 = int.Parse(Console.ReadLine());
19. y2 = int.Parse(Console.ReadLine());
20. Console.WriteLine("Периметр прямоугольника = " + (2 \* (((Math.Sqrt((Math.Pow((x2 - x1), 2))))) + (Math.Sqrt((Math.Pow((y2 - y1), 2)))))));
21. Console.WriteLine("Площадь прямоугольника = " + (((Math.Sqrt((Math.Pow((x2 - x1), 2))))) \* (Math.Sqrt((Math.Pow((y2 - y1), 2))))));
22. Console.ReadLine();
23. }}}

Листинг 5 - Задание 5 (Вычисление площади и периметра треугольника по координатам его вершин)

1. using System;
2. using System.Collections.Generic;
3. using System.Linq;
4. using System.Text;
5. using System.Threading.Tasks;
6. namespace Laboratory\_2
7. {
8. class Program
9. {
10. static void Main(string[] args)
11. {
12. int x1, y1, x2, y2, x3, y3;
13. double a, b, c, p, S;
14. Console.WriteLine("Введите координаты 1 вершины");
15. x1 = int.Parse(Console.ReadLine());
16. y1 = int.Parse(Console.ReadLine());
17. Console.WriteLine("Введите координаты 2 вершины");
18. x2 = int.Parse(Console.ReadLine());
19. y2 = int.Parse(Console.ReadLine());
20. Console.WriteLine("Введите координаты 3 вершины");
21. x3 = int.Parse(Console.ReadLine());
22. y3 = int.Parse(Console.ReadLine());
23. a = Math.Sqrt(Math.Pow((x2 - x1), 2) + Math.Pow((y2 - y1), 2));
24. b = Math.Sqrt(Math.Pow((x3 - x2), 2) + Math.Pow((y3 - y2), 2));
25. c = Math.Sqrt(Math.Pow((x1 - x3), 2) + Math.Pow((y1 - y3), 2));
26. p = (a + b + c) / 2; //формула полупериметра
27. S = Math.Sqrt(p \* (p - a) \* (p - b) \* (p - c)); //по формуле герона считаем площадь
28. Console.WriteLine("Площадь: " + S);
29. Console.WriteLine("Периметр:" + (a + b + c));
30. Console.ReadLine();
31. }}}

# **Результаты выполнения программ**

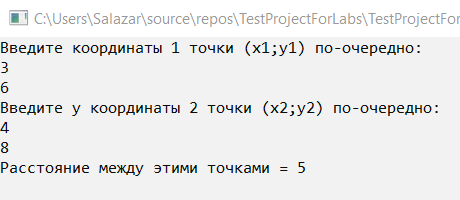


Рисунок 6 — результат выполнения программы 1

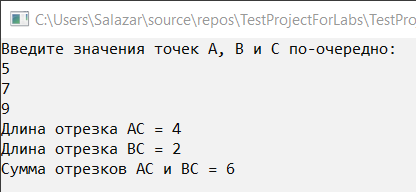


Рисунок 7 — результат выполнения программы 2

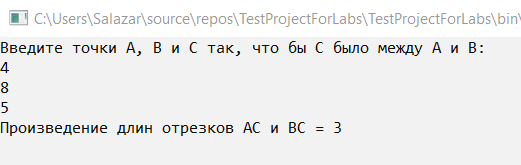


Рисунок 8 — результат выполнения программы 3

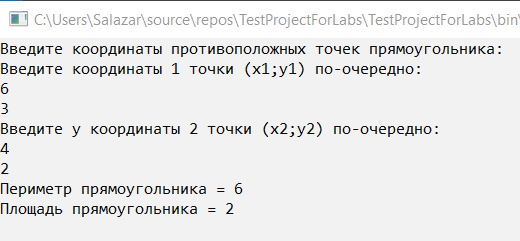


Рисунок 9 — результат выполнения программы 4

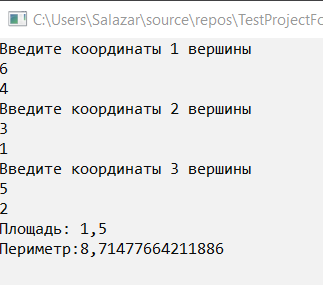


Рисунок 10 — результат выполнения программы 5