3 ПРОЦЕДУРЫ И ФУНКЦИИ – МЕТОДЫ КЛАССА

Задание 1. Разработать методы f(x1, y1, x2, y2), который вычисляет длину отрезка по координатам вершин (x1, y1) и (x2, y2), и метод d(a, b, c), который вычисляет периметр треугольника по длинам сторон a, b, c. С помощью данных методов найти периметр треугольника, заданного координатами своих вершин.

```
Листинг программы:
      try
        int x1, y1, x2, y2, x3, y3;
        Console.Write("x1=");
        x1 = Convert.ToInt32(Console.ReadLine());
        Console.Write("y1=");
        y1 = Convert.ToInt32(Console.ReadLine());
        Console.Write("x2=");
        x2 = Convert.ToInt32(Console.ReadLine());
        Console.Write("y2=");
        y2 = Convert.ToInt32(Console.ReadLine());
        Console.Write("x3=");
        x3 = Convert.ToInt32(Console.ReadLine());
        Console.Write("y3=");
        y3 = Convert.ToInt32(Console.ReadLine());
        Console. WriteLine(\$"result = {d(f(x1, y1, x2, y2), f(x2, y2, x3, y3), f(x3, y3,
x1, y1))");
      catch (Exception ex)
        Console.WriteLine(ex.Message);
      double f(double x1, double y1, double x2, double y2)
        return Math.Pow((Math.Pow((x2 - x1), 2) + Math.Pow((y2 - y1), 2)), 1 / 2);
```

					УП 2-40 01 01.31ТП.2471.22.03				
Изм.	Лист.	№ докум	Подп.	Дата					
Pas	раб.	Мушинский М.С.				Лит	Лист	Листов	
Пров.		Толочко П.С.			ПРОЦЕДУРЫ И ФУНКЦИИ –				
					МЕТОДЫ КЛАССА				
Н.контр.					містоды класса	Гродненский ГКТТиД			
Утв.								, ,	

```
double d(double a, double b, double c)
{
   return a + b + c;
}
```

Таблица 3.1 – Входные и выходные данные

Входные данные	Выходные данные
x1=1	result = 3
y1=2 x2=3	
x2=3	
y2=4 x3=5	
x3=5	
y3=6	

Анализ результатов:

```
x1=1
y1=2
x2=3
y2=4
x3=5
y3=6
result = 3
```

Рисунок 3.1 – Результаты работы программы Источник: собственная разработка

Задание 2 и задание 3. Постройте таблицу значений функции y = f(x) для $x \in [a,b]$ с шагом h. Перегрузите метод f так, чтобы его сигнатура соответствовала виду static void f (double x, out double y).

```
Листинг программы:
class Programm
{
    public static void Main(string[] args)
    {
        try
        {
            Console.Write("введит a = ");
            var a = double.Parse(Console.ReadLine());
            Console.Write("введит b = ");
```

Изм	Лист	№ докум.	Подп.	Дата

```
var b = double.Parse(Console.ReadLine());
    Console.Write("введит h = ");
    var h = double.Parse(Console.ReadLine());
    double y;
    for (double i = a; i \le b; i += h)
       Task(i, out y);
       Console.WriteLine(f{Math.Round(i, 1)} = \{y\});
     }
  catch (Exception ex)
    Console.WriteLine(ex.Message);
static double Task(double x)
  double y = 0;
  if (x > 0)
     y = x;
  else if (x \ge -1 \&\& x \le 0)
     y = 0;
  else
     y = Math.Pow(x, 2);
  return y;
static void Task(double x, out double y)
  y = 0;
  if (x > 0)
     y = x;
```

№ докум.

Изм Лист

Подп. Дата

```
}
else if (x >= -1 && x <= 0)
{
    y = 0;
}
else
{
    y = Math.Pow(x, 2);
}
}</pre>
```

Таблица 3.2 – Входные и выходные данные

Входные данные	Выходные данные
a = 10	f10 = 10
b = 20	f11 = 11
h = 1	f12 = 12
	f13 = 13
	f14 = 14
	f15 = 15
	f16 = 16
	f17 = 17
	f18 = 18
	f19 = 19
	f20 = 20

Анализ результатов:

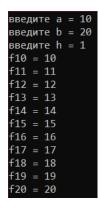


Рисунок 3.2 — Результаты работы программы Источник: собственная разработка

						Лист
					УП 2-40 01 01.31ТП.2471.22.03	
Изм	Лист	№ докум.	Подп.	Дата		