



Департамент образования Ярославской области  
Государственное профессиональное образовательное  
автономное учреждение Ярославской области  
«ЯРОСЛАВСКИЙ ПРОМЫШЛЕННО-ЭКОНОМИЧЕСКИЙ КОЛЛЕДЖ им.  
Н.П. ПАСТУХОВА»

**ОТЧЕТ О ЛАБОРАТОРНЫХ РАБОТАХ**  
**по учебной дисциплине**  
**«Основы алгоритмизации и программирования»**  
ОЛР 090207.09

Студент

\_\_\_\_\_ В.О. Терентьева

« \_\_\_\_ » \_\_\_\_\_ 2022 \_\_\_\_ г.

Преподаватель

\_\_\_\_\_ Е.С. Шестёркина

« \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

## Содержание

Практическая работа № 1,2.....	3
Теоретическая часть.....	3
Практическая часть.....	4
Задача 1.....	4
Задача 2.....	6
Задача 3.....	8

## **Практическая работа № 1,2.**

**Тема:** Составление блок-схем линейных и разветвляющихся и циклических алгоритмов.

**Цель:** Научиться составлять блок-схемы линейных, разветвляющихся и циклических алгоритмов

- 1) Выполнить этапы 1-3 для решения задач.
- 2) Построить алгоритм, используя программу MS Visio.
- 3) Для циклических алгоритмов построить трассировочную таблицу на одном из тестовых наборов

### **Теоретическая часть**

Этапы решения задачи на ЭВМ:

1. Постановка задачи (должно быть четко сформулировано, что дано и что требуется найти, определить полный набор исходных данных).
2. Формализация задачи (задача переводится на язык математических формул, уравнений, отношений).
3. Построение алгоритма.
4. Составление программы на языке программирования.
5. Отладка и тестирование программы.
6. Проведение расчетов и анализ полученных результатов.

Базовые алгоритмические структуры:

- следование,
- ветвление,
- повторение.

На рисунке 1 представлены условные обозначения, используемые при записи алгоритма с помощью блок-схем, для указанных структур: для следования, ветвления, выбора (частный случай ветвления), для повторения (цикл с предусловием, цикл с постусловием, цикл с параметром).

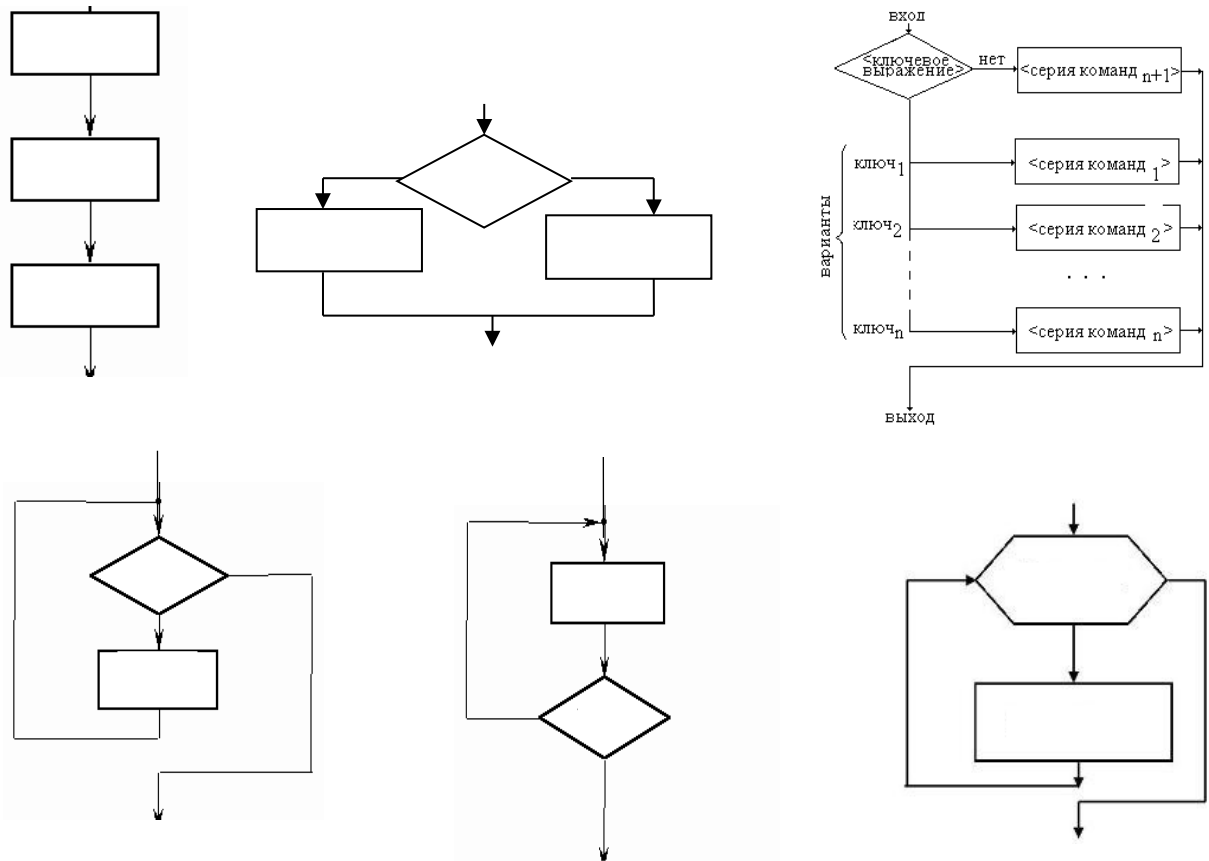


Рисунок 1. Условные обозначения, используемые при записи алгоритма с помощью блок-схем.

## Практическая часть

Вариант №24

### Задача 1

Определить, сколько времени в минутах затратит школьник на дорогу из школы до стадиона, если известна длина этого расстояния  $S$  км и средняя скорость движения школьника  $v$  км/ч?

I. Вход:  $S$  (вещественное число, расстояние от школы до стадиона);  $v$  (вещественное число, средняя скорость движения школьника). Ввод с клавиатуры.

Выход:  $t$  (вещественное число, время, затраченное на дорогу в минутах) или  $t1$  (вещественное число, время, затраченное на дорогу в часах) и  $t2$  (вещественное число, время, затраченное на дорогу в минутах).

Вывод на экран

II. Формализация

$S$  – км

$v$  – км/ч

$t = S/v$

$t, S, v > 0$  – число

$t, S, v < 0$  – нет решений

$v = 0$  – нет решений

$t = t * 60$  – в минутах; если  $t > 60$ , то  $t1 = t/60$  (часов),  $t2 = t - t1 * 60$  (минут)

### III. Блок-схема

На рисунке 2 представлена блок-схема для вычисления времени, которое затратит школьник на путь от школы до стадиона.

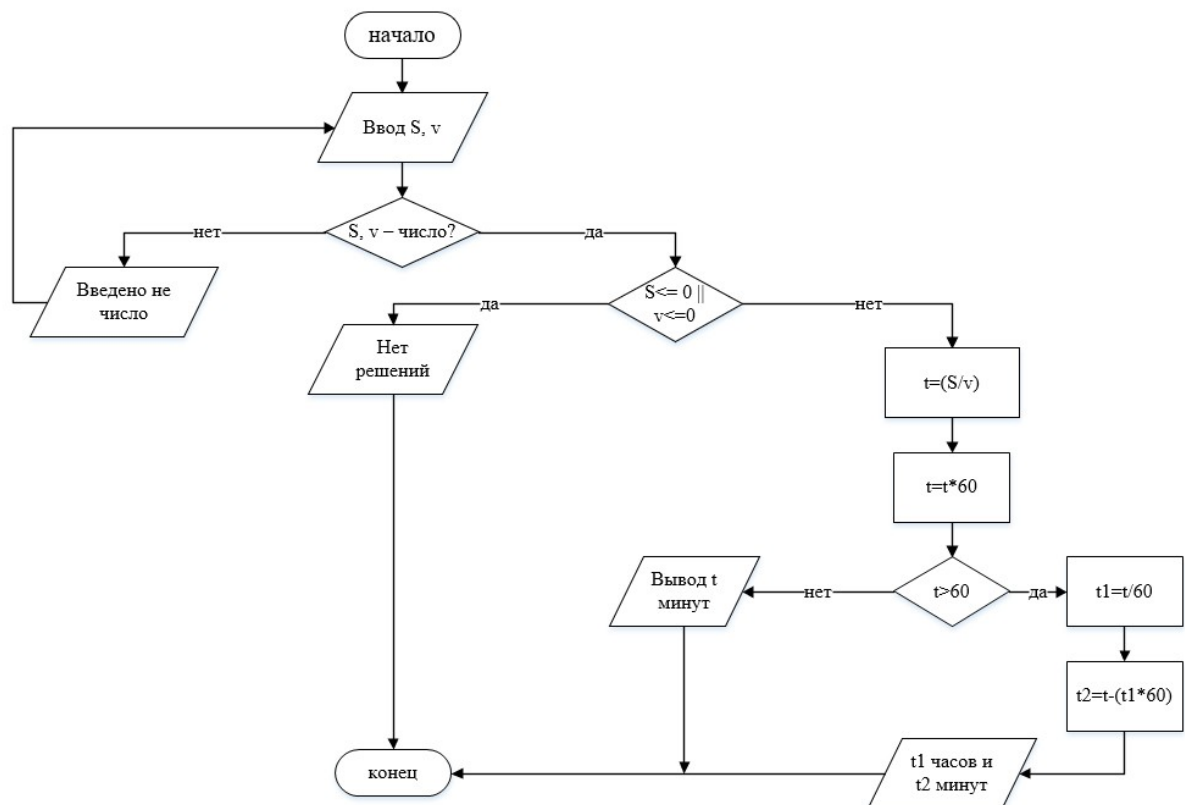


Рисунок 2. Блок-схема для вычисления времени, которое затратит школьника на путь от школы до стадиона.

### IV. Программа

На рисунке 3 представлена программа для определения, затраченного

```

7 namespace ConsoleApp1
8 {
9     class Program
10    {
11        static void Main()
12        {
13            try
14            {
15                Console.WriteLine("Введите расстояние s в км");
16                double s = Convert.ToDouble(Console.ReadLine());
17
18                Console.WriteLine("Введите скорость v в км/ч");
19                double v = Convert.ToDouble(Console.ReadLine());
20                double t; double t1; double t2;
21
22                if (s <= 0 || v<=0)
23                {
24                    Console.WriteLine("Значение должно быть больше 0");
25                }
26                else
27                {
28                    t = (s / v)*60;
29                    if (t > 60)
30                    {
31                        t1 = s / v;
32                        t1 = Math.Truncate(t1);
33                        t2 = t % 60;
34                        Console.WriteLine("Расстояние: {0}ч. {1}мин.", t1, t2);
35                    }
36                    else
37                    {
38                        Console.WriteLine("Расстояние: {0}мин.", t);
39                    }
40                }
41            }
42            catch
43            {
44                Console.WriteLine("Введено не число");
45            }
46            Console.ReadLine();
47        }
48    }
49 }

```

времени (в минутах) школьником на дорогу из школы до стадиона.

V.

### Тестирование

В таблице 1 представлены тестовые наборы для проверки работоспособности программы, выполненной в предыдущем шаге.

Таблица 1. Тестовые наборы для 1 задачи

N	Вход		Выход				результат
	S	v	t	t1	t2	сообщение	

нормальное	11	5	-	2	12	-	2 часа 12 минут
нормальное	4	3	-	1	20	-	1 час 20 минут
нормальное	0,5	5	6	-	-	-	6 минут
экстремальное	3	0	-	-	-	Нет решений	-
экстремальное	1	-2	-	-	-	Нет решений	-
экстремальное	-3	2	-	-	-	Нет решений	-
исключительно е	?	1	-	-	-	Введено не число	-
исключительно е	3	ь	-	-	-	Введено не число	-

На рисунках 6-8 представлены скриншоты окна консоли с работающей программой со значениями из таблицы тестирования.

```

Y:\Student\21ИП1+21ИП2\ОАИП\Терентьева Виктория\ConsoleApp1\bin\Debug\ConsoleApp1.exe
Введите расстояние s в км
11
Введите скорость v в км/ч
5
Расстояние: 2ч. 12мин.

```

Рисунок 4. Окно консоли с нормальными значениями из таблицы тестирования

```

Y:\Student\21ИП1+21ИП2\ОАИП\Терентьева Виктория\ConsoleApp1\bin\Debug\ConsoleApp1.exe
Введите расстояние s в км
1
Введите скорость v в км/ч
-2
Значение должно быть больше 0

```

Рисунок 5. Окно консоли с экстремальными значениями из таблицы тестирования

```

Y:\Student\21ИП1+21ИП2\ОАИП\Терентьева Виктория\ConsoleApp1\bin\Debug\ConsoleApp1.exe
Введите расстояние s в км
3
Введите скорость v в км/ч
ь
Введено не число

```

Рисунок 6. Окно консоли с исключительными значениями из таблицы тестирования

## VI. Вывод

Мы написали корректно работающую программу для задачи, по поиску времени, которое школьник затратил на путь от школы до стадиона.

### Задача 2

Определить стоимость заказа печати фотографий. Исходные данные для расчета— размер (9x12, 12x15 или 18x24) и количество фотографий. Если количество фотографий больше 20, то предоставляется скидка 10%.

I. Вход: 1 (фотография размером 9x12), 2 (фотография размером 12x15), 3 (фотография размером 18x24);  $t$  (вещественное число, количество фотографий) вводится с клавиатуры.

Выход:  $x$  (вещественное число, стоимость заказа).

Вывод на экран

### II. Формализация

Если  $a_1 \Rightarrow x=14$ ,  $a_2 \Rightarrow x=16$ ,  $a_3 \Rightarrow x=20$

$t > 0$  - число

$t < 0$  – нет решений

$t = 0$  – нет решений

$Sum = t * x$ , если  $t > 20$ , то  $Sum = sum - 0.1 * sum$

### III. Блок-схема

На рисунке 3 представлена блок-схема, для определения стоимости фотографии.



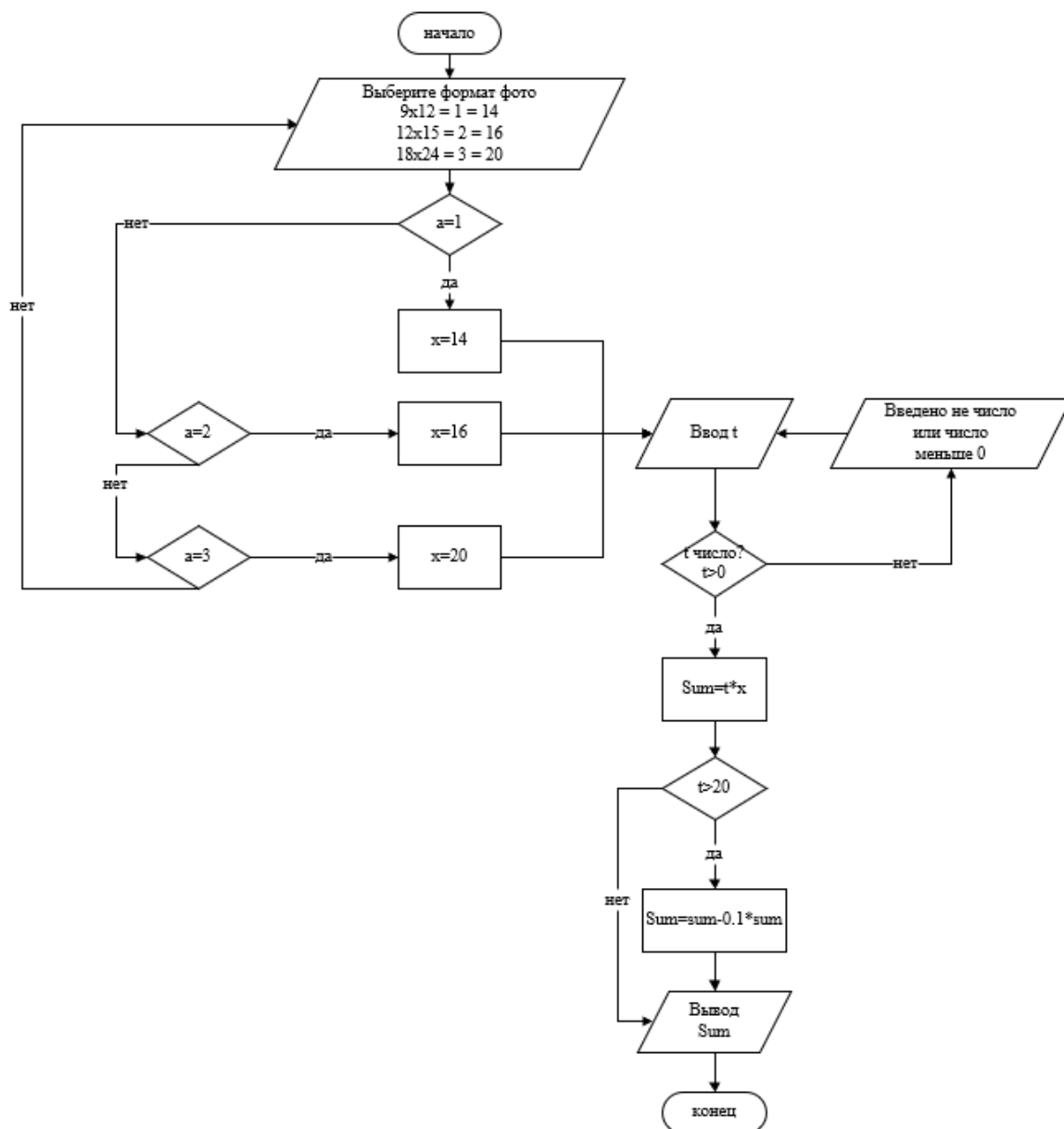


Рисунок 7. Блок-схема для вычисления стоимости фотографии.

#### IV. Программа

#### V. Тестирование

В таблице 2 представлены тестовые наборы для проверки работоспособности программы, выполненной в предыдущем шаге.

Таблица 2. Тестовые наборы для 2 задачи

N	Вход				Выход		Результат
	1	2	3	t	x	сообщение	
нормальное	1	16	20	4	56	-	56

	<b>4</b>						
нормальное	1 4	<b>16</b>	20	20	320	-	320
нормальное	1 4	16	<b>20</b>	23	414	-	414
экстремальное	1 4	16	20	0	-	Нет решений	-
экстремальное	1 4	16	20	-6	-	Нет решений	-
исключительное	1 4	16	20	T	-	Введено не число	-
исключительное	1 4	16	20	^	-	Введено не число	-

## VI. Вывод

Мы написали корректно работающую программу для задачи, по определению стоимости заказа печати фотографии.

### Задача 3

Вывести таблицу значений функции  $y=x^2+\ln x$  с шагом  $h$  на отрезке  $[a,b]$  ( $h,a,b$  вводятся с клавиатуры)

I. Вход:  $x$  (вещественное число).  $h$  (вещественное число, шаг на отрезке),  $a$  (вещественное число, начало отрезка)  $b$  (вещественное число, конец отрезка)> ввод с клавиатуры.

Выход: таблица значений функции  $y=x^2 +\ln x$  – пары  $[a,b]$ . Вывод на экран.

#### II. Формализация

$y=x^2 +\ln x$  – возрастающая функция

$[a,b]$  – отрезок

$a$  – начало,  $b$  – конец.

$a, b, h$  – числа

$a \leq b$

#### III. Блок-схема

На рисунке 4 представлена блок-схема, для вывода таблицы значений функции  $y=x^2+\ln x$  с шагом  $h$  на отрезке  $[a,b]$ .

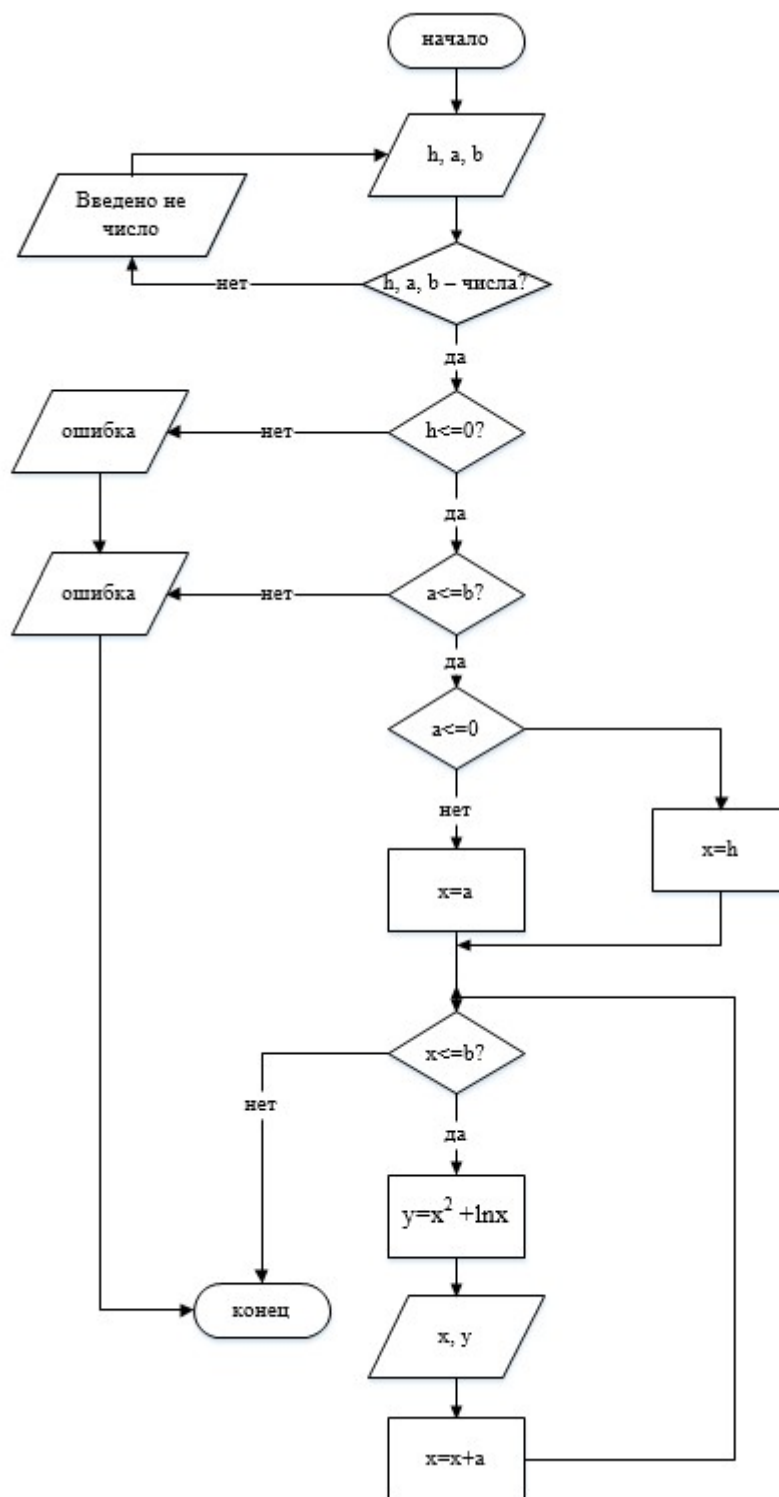


Рисунок 8. Блок-схема для вывода таблицы значений функции  $y=x^2+\ln x$  с шагом  $h$  на отрезке  $[a,b]$ .

#### IV. Программа

На рисунке 5 представлены скриншоты программы, для вывода

таблицы значений функции  $y=x^2+\ln x$  с шагом  $h$  на отрезке  $[a,b]$ .

```

7 namespace ConsoleApp1
8 {
9     ссылка: 0
10    class Program
11    {
12        ссылка: 0
13        static void Main()
14        {
15            Console.WriteLine("Введите начальное значение отрезка");
16
17            double a = double.Parse(Console.ReadLine());
18            Console.WriteLine("Введите конечное значение отрезка");
19            double b = double.Parse(Console.ReadLine());
20            Console.WriteLine("Введите шаг");
21            double h = double.Parse(Console.ReadLine());
22            double x; double y;
23
24            if (h <= 0)
25            {
26                Console.WriteLine("Шаг должен быть положительным");
27            }
28            else
29            {
30                if (a <= b)
31                {
32                    if (a <= 0)
33                    {
34                        x = h;
35                        while (x <= b)
36                        {
37                            y = x*x + Math.Log(x);
38                            Console.WriteLine("{0,9}|{1:F4}|", x, y);
39                            x = x + h;
40                        }
41                    }
42                    else
43                    {
44                        x = a;
45                        while (x <= b)
46                        {
47                            y = x*x + Math.Log(x);
48                            Console.WriteLine("{0,9}|{1:F4}|", x, y);
49                            x = x + h;
50                        }
51                    }
52                }
53                else
54                {
55                    Console.WriteLine("Начальное значение должно быть меньше или равно конечному");
56                }
57            }
58        }
59    }
60    catch
61    {
62        Console.WriteLine("Введено не число");
63    }
64    Console.ReadLine();
65 }
66 }
67 }

```

Рисунок 9. Программа для 3 задачи

## V. Тестирование

В таблице 3 представлены тестовые наборы для проверки работоспособности программы, выполненной в предыдущем шаге.

Таблица 3. Тестовые наборы для 3 задания

N	Вход			Выход			Результат
	h	a	b	x	y	сообщение	
нормальное	0.5	2	6	2	4,6931	-	2 4,6931
				2,5	5,9163		2,5 5,9163
				3	7,0986		3 7,0986
				3,5	8,2528		3,5 8,2528
				4	9,3863		4 9,3863
				4,5	10,5041		4,5 10,5041
				5	11,6094		5 11,6094
				5,5	12,7047		5,5 12,7047
экстремальное	1	-2	9	1	1,0000	-	1 1,0000
				2	4,6931		2 4,6931
				3	10,0986		3 10,0986
				4	17,3863		4 17,3863
				5	26,6094		5 26,6094
				6	37,7918		6 37,7918
				7	50,9459		7 50,9459
				8	66,0794		8 66,0794
экстремальное	-1	-2	2	-	-	Шаг должен быть положительным	-
				искл	2		4
искл	1	ь	3	-	-	Введено не число	-

На рисунках 10-11 представлены скриншоты окна консоли с работающей программой со значениями из таблицы тестирования.

```
Y:\Student\21ИП1+21ИП2\ОАИП\Терентьева Виктория\ConsoleApp3\ConsoleApp3\bin\Debug\ConsoleApp3.exe
Введите начальное значение отрезка
2
Введите конечное значение отрезка
6
Введите шаг
0,5
|      2 | 4,6931 |
|     2,5 | 7,1663 |
|      3 | 10,0986 |
|     3,5 | 13,5028 |
|      4 | 17,3863 |
|     4,5 | 21,7541 |
|      5 | 26,6094 |
|     5,5 | 31,9547 |
|      6 | 37,7918 |
```

Рисунок 10. Окно отладки с нормальными значениями из таблицы тестирования

```
Y:\Student\21ИП1+21ИП2\ОАИП\Терентьева Виктория\ConsoleApp3\ConsoleApp3\bin\Debug\ConsoleApp3.exe
Введите начальное значение отрезка
-2
Введите конечное значение отрезка
2
Введите шаг
-1
Шаг должен быть положительным
```

Рисунок 11. Окно отладки с экстремальными значениями из таблицы тестирования

```
Y:\Student\21ИП1+21ИП2\ОАИП\Терентьева Виктория\ConsoleApp3\ConsoleApp3\bin\Debug\ConsoleApp3.exe
Введите начальное значение отрезка
2
Введите конечное значение отрезка
4
Введите шаг
!
Введено не число
```

Рисунок 12. Окно отладки с исключительными значениями из таблицы тестирования

## VI. Вывод

Мы написали корректно работающую программу для задачи, по выводу таблицы значений функции  $y=x^2+\ln x$  с шагом  $h$  на отрезке  $[a,b]$ .

**Вывод:** Мы научились составлять блок-схемы линейных, разветвляющих и циклических алгоритмов и писать корректно работающие программы к этим задачам.