Задание 1

namespace Lab7

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

// Функция для нахождения минимального элемента в массиве

static int min\_elem(int[] arr, int n)

{

if (n == 0)

{

throw new ArgumentException("Массив не должен быть пустым.");

}

int min = arr[0];

for (int i = 1; i < n; i++)

{

if (arr[i] < min)

{

min = arr[i];

}

}

return min;

}

// Функция для преобразования строки в массив целых чисел

static int[] ParseArray(string input)

{

return input.Split(' ').Select(int.Parse).ToArray();

}

private void button1\_Click(object sender, EventArgs e)

{

int na = int.Parse(textBoxNa.Text);

int[] a = ParseArray(textBoxA.Text);

int nb = int.Parse(textBoxNb.Text);

int[] b = ParseArray(textBoxB.Text);

int nc = int.Parse(textBoxNc.Text);

int[] c = ParseArray(textBoxC.Text);

textBoxMinA.Text = min\_elem(a, na).ToString();

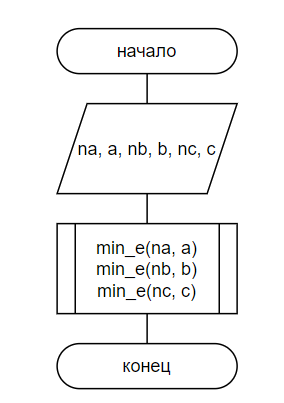
textBoxMinB.Text = min\_elem(b, nb).ToString();

textBoxMinC.Text = min\_elem(c, nc).ToString();

}

}

}



Схнма основной логики

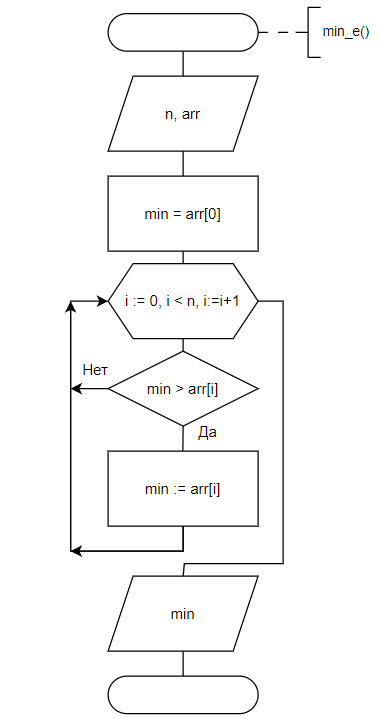
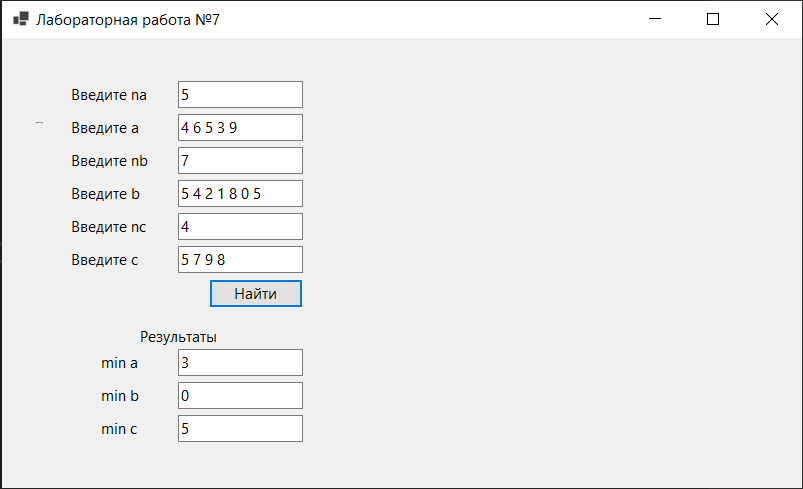


Схема функции min\_e()



Задание 2

namespace Lab7

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

// Функция для вычисления суммы S

static double CalculateSum(double x, int n)

{

double sum = 0;

for (int i = 0; i <= n; i++)

{

sum += (Math.Pow(Math.Log(3), i) / Factorial(i)) \* Math.Pow(x, i);

}

return sum;

}

// Рекурсивная функция для вычисления факториала

static double Factorial(int number)

{

if (number == 0)

return 1;

return number \* Factorial(number - 1);

}

// Функция для преобразования строки в массив целых чисел

static int[] ParseArray(string input)

{

return input.Split(' ').Select(int.Parse).ToArray();

}

private void button1\_Click(object sender, EventArgs e)

{

double xStart = 0.1;

double xEnd = 1.0;

int n = 10;

double step = (xEnd - xStart) / 10;

textBoxXStart.Text = xStart.ToString();

textBoxXEnd.Text = xEnd.ToString();

textBoxN.Text = n.ToString();

textBoxStep.Text = step.ToString();

for (int i = 0; i <= 10; i++)

{

double x = xStart + i \* step;

double y = Math.Pow(3, x);

double s = CalculateSum(x, n);

textBoxResult.Text += $"x = {x:F2}, Y = {y:F6}, S = {s:F6}" + Environment.NewLine;

}

}

}

}

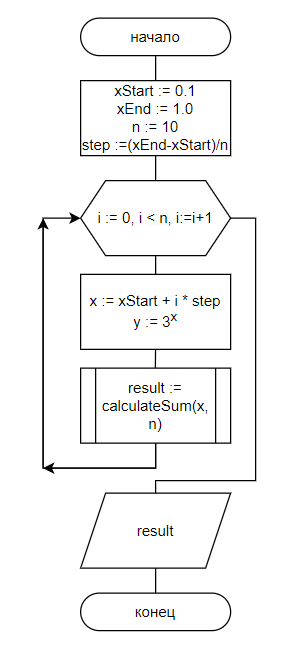


Схема основного алгоритма

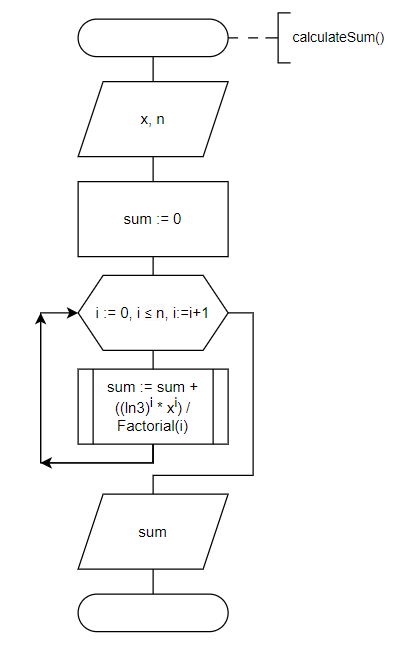


Схема функции calculateSum()

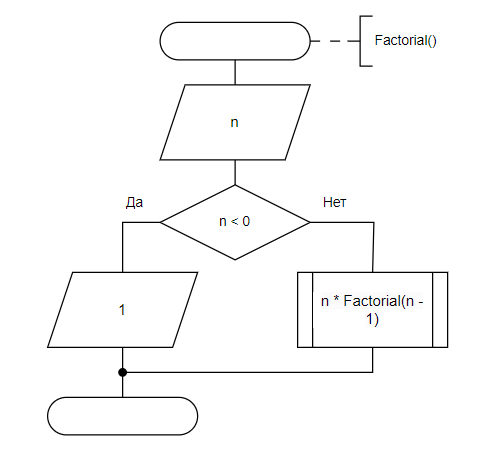


Схема функции Factorial()

