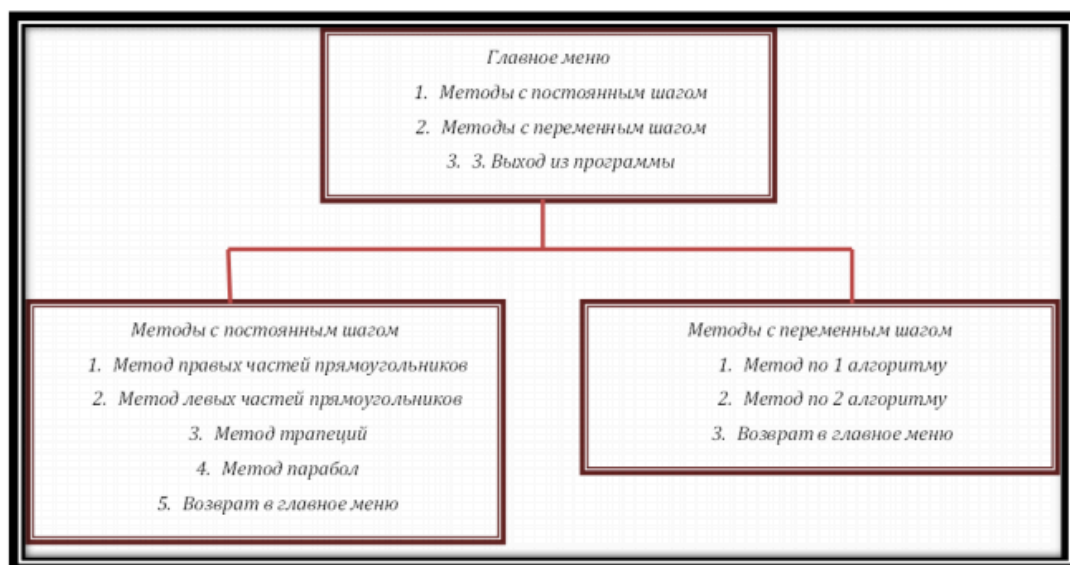


Лабораторная работа №1: Численное интегрирование

Постановка задачи: составить программу, которая реализует методы численного интегрирования с постоянным и переменным шагом для интеграла:

$$\int_0^1 e^{-x^2} dx$$

Программа должна реализовывать следующее меню:



Мат модель:

$$S \approx \sum_{i=0}^{n-1} y_i \Delta x_i$$

Код программы:

```
namespace ConsoleApp
{
    class Program
```

```

{
    static void Main()
    {
        var x = 0;
        while (x != 3)
        {
            Console.Clear();
            Console.WriteLine("Я вас категорически приветствую, вам доступны
следующие функции:");
            Console.WriteLine("1. Методы с постоянным шагом.");
            Console.WriteLine("2. Методы с переменным шагом.");
            Console.WriteLine("3. Выход из программы");
            x = ii();
            switch (x)
            {
                case 1:
                    const_step();
                    break;
                case 2:
                    variable_step();
                    break;
                case 3:
                    Console.WriteLine("До свидания!");
                    break;
                default:
                    Console.WriteLine("Неверная команда.");
                    alert();
                    break;
            }
            if (s != 0)
            {
                output(s);
                s = 0;
                alert();
            }
        }
    }
}

static void const_step()
{
    Console.Clear();
    Console.WriteLine("1. Метод правых частей прямоугольника.");
    Console.WriteLine("2. Метод левых частей прямоугольника.");
    Console.WriteLine("3. Метод трапеций.");
    Console.WriteLine("4. Метод парабол.");
    Console.WriteLine("5. Вызврат в главное меню.");
    var x = 0;
    x = ii();
    switch (x)
    {
        case 1:
            r_sq();
            break;
        case 2:
            l_sq();
            break;
        case 3:
            trap();
            break;
        case 4:
            par();
            break;
        case 5:
            break;
        default:
            Console.WriteLine("Неверная команда.");
            alert();
    }
}

```

```

        const_step();
        break;
    }
}
static void variable_step()
{
    Console.Clear();
    Console.WriteLine("1. Метод по 1 алгоритму.");
    Console.WriteLine("2. Метод по 2 алгоритму.");
    Console.WriteLine("3. Вызват в главное меню.");
    var x = 0;
    x = ii();
    switch (x)
    {
        case 1:
            alg_1();
            break;
        case 2:
            alg_2();
            break;
        case 3:
            break;
        default:
            Console.WriteLine("Неверная команда.");
            alert();
            variable_step();
            break;
    }
}
0, I2n = 0;
public static double a, b, s = 0, i, h, n, s1 = 0, s2 = 0, e = 0, r, In =
static void r_sq()
{
    Console.Clear();
    inp();
    i = a + h;
    while (i <= b)
    {
        s = s + func(i);
        i = i + h;
    }
    s = h * s;
}
static void l_sq()
{
    Console.Clear();
    inp();
    i = a;
    while (i <= b - h)
    {
        s = s + func(i);
        i = i + h;
    }
    s = h * s;
}
static void trap()
{
    Console.Clear();
    inp();
    double y1=0, y2=0;
    y1 = y1 + func(a);
    y2 = y2 + func(b);
    i = a + h;
    while (i <= b - h)
    {
        s = s + func(i);

```

```

        i = i + h;
    }
    s = h * ((y1 + y2) / 2 + s);
}
static void par()
{
    Console.Clear();
    inp();
    s = func(a);
    s = s + func(b);
    i = a + h;
    while (i <= b - h)
    {
        s1 = s1 + func(i);
        i = i + 2 * h;
    }
    i = a + h * 2;
    while (i <= b - h * 2)
    {
        s2 = s2 + func(i);
        i = i + 2 * h;
    }
    s = (h * (4 * s1 + 2 * s2 + s) / 3);
}
static void alg_1()
{
    Console.Clear();
    r = 0;
    v_inp();
    do
    {
        s = 0; i = a;
        while (i <= b - h)
        {
            s = s + func(i);
            //Console.WriteLine("i:" + Math.Exp(Math.Pow(-i, 2)));
            i = i + h;
        }
        s = s * h;
        I2n = s;
        r = Math.Abs(In - I2n);
        In = I2n;
        h = h / 2;
    } while (r > e);
    Console.WriteLine("h = " + h * 2);
}
static void alg_2()
{
    Console.Clear();
    v_inp();
    r = 0; s = 0; i = a; s1 = 0;
    do
    {
        i = i + h / 2;
        while(i<=b-h)
        {
            s1 = s1 + func(i);
            i = i + h;
        }
        s1 = s1 * h;
        I2n = s1;
        r = Math.Abs(In - I2n);
        In = I2n;
        h = h / 2;
        s = s + s1;
    } while (r > e);
}

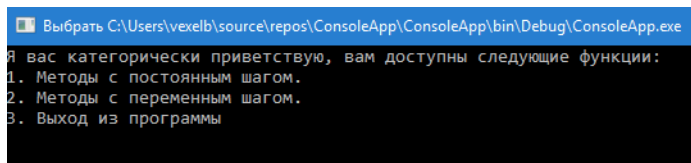
```

```

        Console.WriteLine("h = " + h * 2);
    }
    static void alert()
    {
        Console.WriteLine("Для продолжения нажмите enter,");
        Console.Read();
    }
    static void output(double q)
    {
        Console.WriteLine("Значение: " + q);
    }
    static float ti()
    {
        var q = "0"; float w;
        do
        {
            q = Console.ReadLine();
            if (float.TryParse(q, out w))
                w = float.Parse(q);
            else
                Console.Write("Введите корректное значение:");
        } while (float.TryParse(q, out w) == false);
        return w;
    }
    static void inp()
    {
        Console.Write("Нижняя граница:");
        a = ti();
        Console.Write("Верхняя граница:");
        b = ti();
        Console.Write("Кол-во точек:");
        n = ti();
        h = (b - a) / n;
    }
    static void v_inp()
    {
        Console.Write("Нижняя граница:");
        a = ti();
        Console.Write("Верхняя граница:");
        b = ti();
        Console.Write("Кол-во точек:");
        n = ti();
        h = (b - a) / n;
        Console.Write("Эпсилон:");
        e = ti();
    }
    static int ii()
    {
        var q = "0"; int w;
        do
        {
            q = Console.ReadLine();
            if (int.TryParse(q, out w))
                w = int.Parse(q);
            else
                Console.Write("Введите корректное значение:");
        } while (int.TryParse(q, out w) == false);
        return w;
    }
    static double func(double x)
    {
        return Math.Exp(-(x * x));
    }
}
}

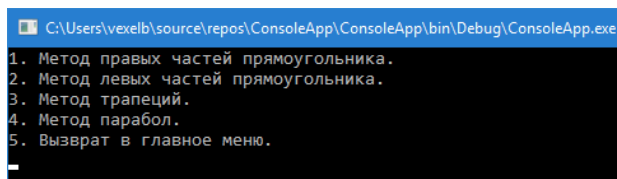
```

Отчет о работе программы:



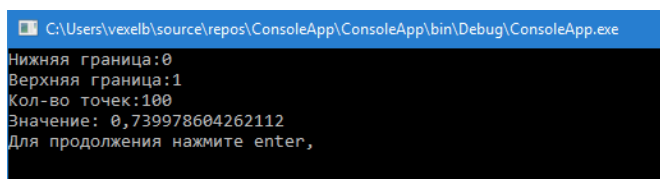
```
C:\Users\velb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Я вас категорически приветствую, вам доступны следующие функции:
1. Методы с постоянным шагом.
2. Методы с переменным шагом.
3. Выход из программы
```

Рис. 1 – основное меню



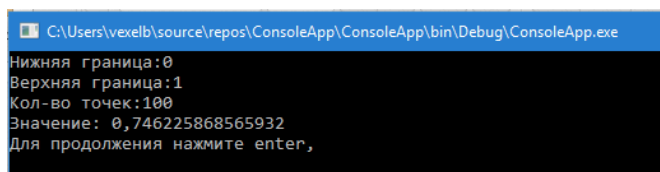
```
C:\Users\velb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
1. Метод правых частей прямоугольника.
2. Метод левых частей прямоугольника.
3. Метод трапеций.
4. Метод парабол.
5. Вызврат в главное меню.
```

Рис. 2 – меню «методы с постоянным шагом»



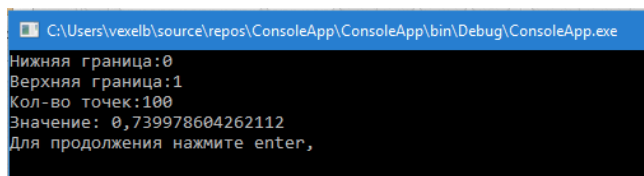
```
C:\Users\velb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Нижняя граница:0
Верхняя граница:1
Кол-во точек:100
Значение: 0,739978604262112
Для продолжения нажмите enter,
```

Рис. 3 – результат работы метода правых частей



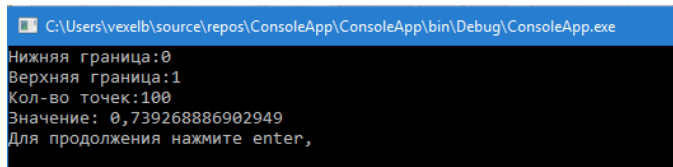
```
C:\Users\velb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Нижняя граница:0
Верхняя граница:1
Кол-во точек:100
Значение: 0,746225868565932
Для продолжения нажмите enter,
```

Рис. 4 – результат работы метода правых частей



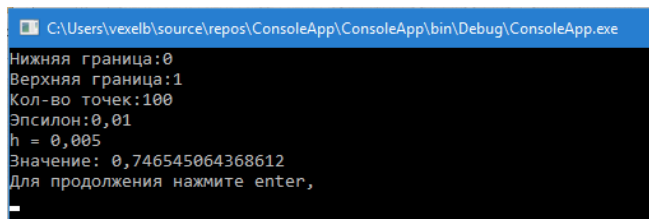
```
C:\Users\velb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Нижняя граница:0
Верхняя граница:1
Кол-во точек:100
Значение: 0,739978604262112
Для продолжения нажмите enter,
```

Рис. 5 – результат работы метода трапеций



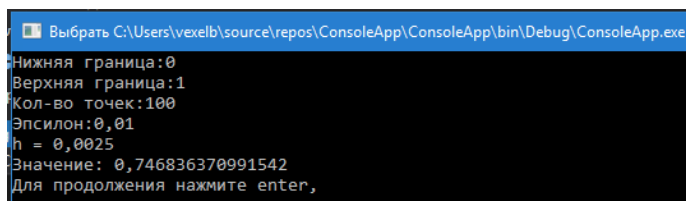
```
C:\Users\vxelb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Нижняя граница:0
Верхняя граница:1
Кол-во точек:100
Значение: 0,739268886902949
Для продолжения нажмите enter,
```

Рис. 6 – результат работы метода парабол



```
C:\Users\vxelb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Нижняя граница:0
Верхняя граница:1
Кол-во точек:100
Эпсилон:0,01
h = 0,005
Значение: 0,746545064368612
Для продолжения нажмите enter,
_
```

Рис. 7 – результат работы 1 алгоритма с переменным шагом



```
Выбрать C:\Users\vxelb\source\repos\ConsoleApp\ConsoleApp\bin\Debug\ConsoleApp.exe
Нижняя граница:0
Верхняя граница:1
Кол-во точек:100
Эпсилон:0,01
h = 0,0025
Значение: 0,746836370991542
Для продолжения нажмите enter,
```

Рис. 8 – результат работы 2 алгоритма с переменным шагом