

*СПбНИУ ИТМО  
Кафедра ИПМ*

*Лабораторная работа №2  
по дисциплине  
«Вычислительная математика»  
«Вычисление определённого интеграла  
методом Симпсона, с заданной точностью»*

*Выполнил  
Широков О.И  
гр.2120*

*Санкт-Петербург  
г.2013*

### 1. Описание метода.

Вычисление определённого интеграла методом Симпсона базируется на разбиении промежутка интегрирования на отрезки и интерполяции функции на каждом отрезке многочленом второй степени по формуле

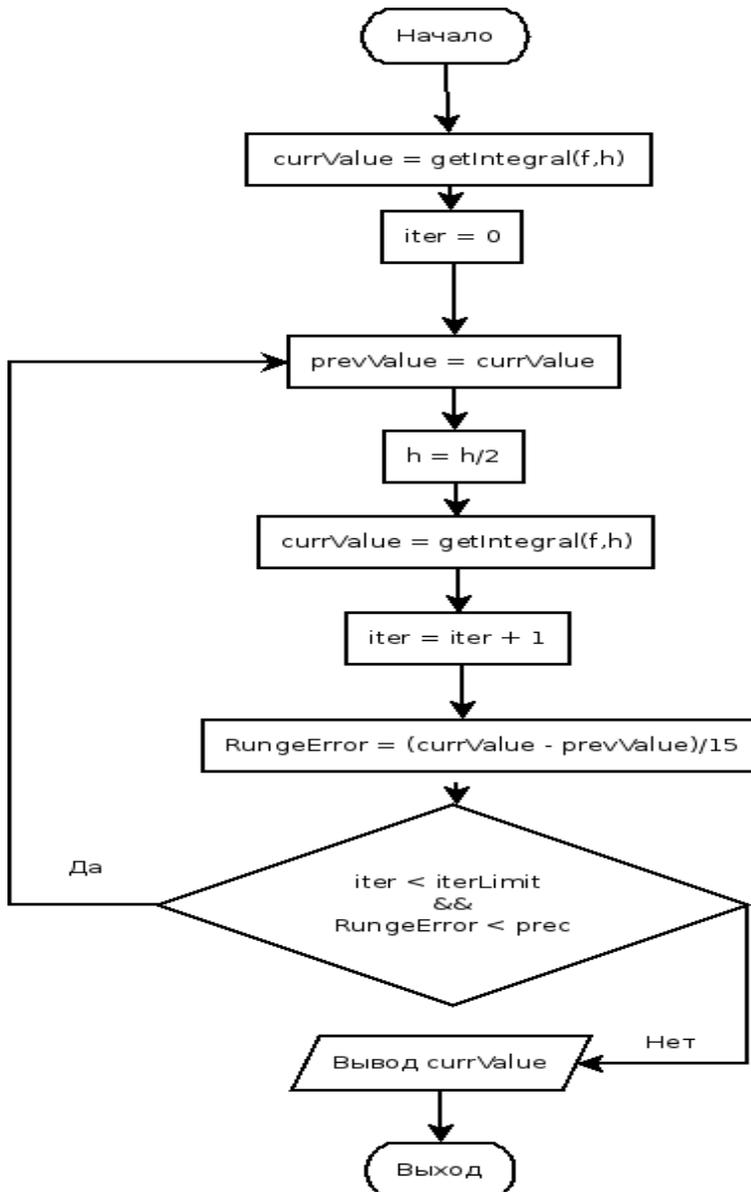
Для промежутка  $[a; b]$

$$\int_a^b f(x) dx = \frac{(b-a)}{6} (f(a) + 4 * f(a+h/2) + f(b)), \text{ где } h - \text{длина отрезка } [a; b]$$

Оценка погрешности выполняется вычислением оценки Рунге, по формуле

$$R = \frac{I_h - I_{h/2}}{15}, \text{ где } I_h \text{ и } I_{h/2} \text{ значения определённых интегралов вычисленных с шагом } h \text{ и шагом } h/2 \text{ соответственно}$$

### 2. Блок схема реализованного метода.



### 3. Исходный текст программы.

*//Файл SolveIntegral.java*

```
package SimpsonMethod;

public class SolveIntegral
{
    IExpression integrableFunction;
    final int iterationLimit;

    public SolveIntegral(IExpression function,int iterationLimit)
    {
        this.integrableFunction = function;
        this.iterationLimit = iterationLimit;
    }

    public double Solve(double a,double b,double precision,double i_step)
    {
        double step = i_step;
        double prevValue;
        double currentValue = getIntegral(a,b,step);
        double RungeError;
        int iter = 0;
        do
        {
            prevValue = currentValue;
            step /= 2;
            currentValue = getIntegral(a,b,step);
            RungeError = Math.abs(prevValue - currentValue)/15;
            iter++;
        }while(RungeError > precision && iter < iterationLimit);
        return currentValue;
    }

    double getIntegral(double a,double b,double step)
    {
        double i;
        double integralValue = 0;
        for(i = a; i < b; i+= step ) //TODO Condition
        {
            double left = integrableFunction.getValue(i);
            double med = 4*integrableFunction.getValue(i + step/2 );
            double right = integrableFunction.getValue(i+step);
            integralValue += (step/6)*(left + med + right);
        }
        return integralValue;
    }
}

//Файл Main.java
package SimpsonMethod;

public class Main
{
    public static void main(String[] args)
    {
        SolveIntegral solver = new SolveIntegral(new Expr(),1000);
        System.out.println(solver.Solve(1,2,Math.pow(1,-5),Math.pow(1,-10)));
        return;
    }
}
```

```

    }
}
//Файл Iexpression.java
package SimpsonMethod;

public interface IExpression
{
    String toString();
    double getValue(double x);
}

//Файл Expr.java
package SimpsonMethod;

class Expr implements IExpression
{
    @Override
    public double getValue(double x)
    {
        double ret = Math.pow(x,2); //Пример для вычисления интеграла
//          y=x^2
        return ret;
    }
}

```

#### **4. Примеры вычисления**

*При равных пределах интегрирования*

*0.0*

$\int_3^9 x^3$ ; точность  $10^{-5}$   
233.99999999999994

#### **5. Выводы**

*Во время выполнения лабораторной работы была разработана программа вычисляющая определённый интеграл методом Симпсона, с заданной точностью.*