Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

«Национальный исследовательский Томский политехнический Университет»



Инженерная школа ядерных технологий

01.03.02 «Прикладная математика и информатика»

**Лабораторная работа №3**

по дисциплине:

**Численные методы**

Вариант 14

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Выполнил:** |  | | | | |
| Студент группы | 0В01 |  | Белясов Архип Александрович |
|  |  |  |
| **Проверил:** | Крицкий Олег Леонидович | | | | |
| преподаватель |  |  |  |  |  | |
|  |  |  |  |  |  | |

# Цель работы:

Вычислить определенный интеграл с точностью ε=10-4 методами прямоугольников, Монте-Карло и Симпсона. Сравнить результаты, полученные различными методами.

# Теория:

Зададимся некоторыми  и построим интерполяционный [многочлен](http://scask.ru/f_book_m_cat.php?id=22) *Ln(x)* степени *n - 1*, совпадающий с *f(x)* в точках  . Положим



Имеем



Разность оценим, воспользовавшись оценкой погрешности [интерполяционного многочлена Лагранжа](http://scask.ru/b_book_bcm.php?id=11)



где  . Отсюда



Произведем в последнем интеграле замену переменных, положив  Тогда



где





Таким образом, справедлива оценка

 (1)

Пусть все **  различны. Тогда



После замены переменных получим

 (2)

где

 (3)

Таким образом, построенная квадратурная формула имеет вид

 (4)

**Формула прямоугольников**

В качестве высот прямоугольников здесь принимаются значения функции, вычисленные **в серединах** промежуточных отрезков, и в общем виде формула приближённых вычислений запишется следующим образом:

Где – шаг стандартного «равноотрезочного» разбиения.

Аналогично из формулы (1) получается формула оценки погрешности метода:

**Формула Симпсона**

Формула Симпсона связана с формулой (4) и может быть получена из нее напрямую. Заменим функцию *f*(*x*) на ее многочлен Лагранжа второй степени на интервале [*x2i*-1;*x2i*+1] с дополнительной внутренней точкой интерполирования *x2i*:

.

Проводя выкладки, получаем требуемую формулу:



Аналогично из формулы (1) получается формула оценки погрешности метода:



**Метод Монте-Карло**

Интеграл выражается формулой:



где *x1,...,xi*- точки, равномерно распределенные на заданном интервале, а N - число точек *xi*.

Расчетная формула для погрешности метода Монте-Карло:



Если точность  и гарантийная вероятность  заданы, то выводим необходиоме число N:



**Ход работы:**

Зададим функцию funk

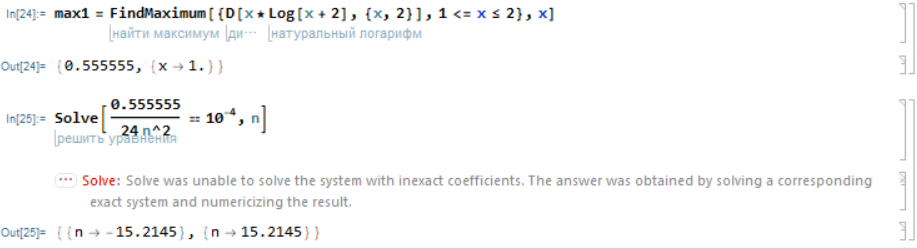
function [f]=funk(x)

f=x\*log(x+2);

end

**Формула прямоугольников:**

Найдем n, необходимое для достижения заданного уровня точности, воспользовавшись оценкой погрешности метода:



Таким образом, n = 16.

Реализация алгоритма в *Matlab* :

function [sum] = pr(a, b, n)

h = (b-a)/n;

sum = 0;

for i = 1:n-1

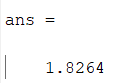
sum = sum + funk(a + h\*i+h/2)\*h;

end

disp(sum)

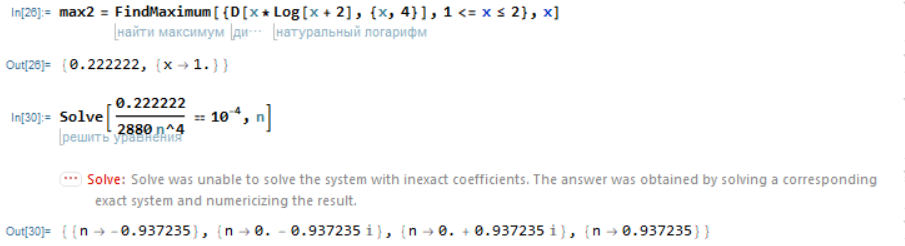
end

Результат вычисления:



**Формула Симпсона:**

Найдем n, необходимое для достижения заданного уровня точности, воспользовавшись оценкой погрешности метода:



Таким образом, n = 1.

Реализация алгоритма в *Matlab*:

function [sum] = simpson(a, b, n)

h = (b-a)/2/n;

sum = (funk(a) + funk(b))\*h/3;

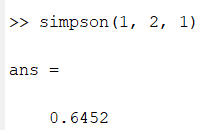
for i = 1:n-1

sum = sum + 4\*funk(a + h\*(2\*i-1))\*h/3 + 2\*funk(a + h\*(2\*i))\*h/3;

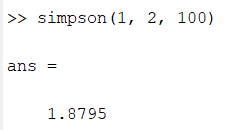
end

end

Результат вычисления:



Результат вычислений в данном случае имеет большую погрешность, что связано со значением n. Увеличим значение n и возьмем n = 100. Результат вычисления интеграла методом Симпсона в этом случае:



**Метод Монте-Карло:**

Найдем n, необходимое для достижения заданного уровня точности, воспользовавшись оценкой погрешности метода:

ε=10-4 , = 0,01, 

Реализация алгоритма в *Matlab*:

function [sum] = MC(a, b, n)

h = (b-a)/n;

sum=0;

for i = 1:n

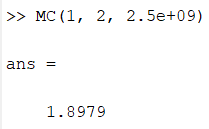
sum = sum + funk(a + h\*(i-1));

end

sum = sum\*(b-a)/n;

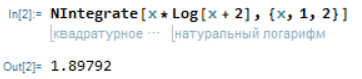
end

Результат вычисления:



Сравним результаты, полученные разными методами:

Значение интеграла, вычесленное встроенной функцией среды *Wolfram*:



Таким образом, наиболее точное решение получено последним из использованных методов.

**Вывод:** в ходе лабораторной работы заданный интеграл был вычислен методами трапеций, Симпсона и Монте-Карло. Наиболее точное решение получено методом Монте-Карло, значение n для которого было наибольшим (2.5\*109).

Реализация алгоритмов была проведена в среде *Matlab* и *Wolfram*.