**Санкт-Петербургский национальный исследовательский университет**

**информационных технологий, механики и оптики**

**Кафедра информатики и прикладной математики**

Вычислительная математика

Лабораторная работа №2

«Интегральные вычисления методом Симпсона с заданной точностью ».

Выполнил Топалов Олег

Группа 2120

Проверил Шипилов П.А.

2013 г.

Задача: Организация вычислительного алгоритма интегрирования методом Симпсона(методом парабол) с заданной точностью.

# Описание метода

***Интегрирование по методу Симпсона.*** Формула трапеций дает результат, сильно зависящий от величины шага *h*, что сказывается на точности вычисления определенного интеграла особенно в тех случаях, когда функция имеет немонотонный характер. Можно предположить повышение точности вычислений, если вместо отрезков прямых, заменяющих криволинейные фрагменты графика функции *f(x)*, использовать, например, фрагменты парабол, проводимых через три соседних точки графика. Подобная геометрическая интерпретация лежит в основе метода Симпсона для вычисления определенного интеграла. Весь интервал интегрирования [*a,b*] разбивается на четное число одинаковых отрезков *n*, длина отрезка также будет равна *h=(b-a)/n.* Формула Симпсона имеет вид:

1. 

В формуле выражения в скобках представляют собой суммы значений подынтегральной функции соответственно на концах нечетных и четных внутренних отрезков.

В нашем случае N не обязательно четное, так как мы будем использовать видоизмененную формулу, описанную ниже.

# Блок-схема алгоритма



# Расчетные формулы

H = (B-A)/N, где A,B-границы интегрирования, N –число разбиений, H – шаг разбиения.

Si = h/6\*(F(Xi) + F(Xi+h) + 4F(Xi+h/2)) – площадь i-того участка интегрирования.
O = |h-h/2|/15 – формула Оценки Рунге для метода Симпсона.

# Программа

1. #include <iostream>
2. #include <cmath>
3. using namespace std;
4.
5. long double F(long double x)
6. {
7. long double degree = x\*M\_PI/180;
8. long double result=(sin(degree) \* cos(degree) + x\*x\*x + 8\*x\*sqrt(x));
9. return result;
10. }
11.
12. int main()
13. {
14. long double e = 1e-6;
15. int N = 1;
16. long double A, B, S1=0, S2=0, Runge;
17. cout<< "Input A:";
18. cin>> A;
19. cout<< "Input B:";
20. cin>> B;
21. do
22. {
23. long double h = (B-A)/double(N);
24. S1=0;
25. S2=0;
26. for(int i=0;i<N;i++)
27. {
28. S1+=h/6\*(F(A+i\*h)+F(A+i\*h+h)+4\*F(A+i\*h+h/2));
29. }
30. N\*=2;
31. h/=2.0;
32. for(long int i=0;i<N;i++)
33. {
34. S2+=h/6\*(F(A+i\*h)+F(A+i\*h+h)+4\*F(A+i\*h+h/2));
35. }
36. Runge = fabs(h-h/2)\*1.0/15;
37. }
38. while(Runge>e);
39. cout<<"E = "<<e<<endl;
40. cout<<"Runge = "<<Runge<<endl;
41. cout<<"N = "<<N<<endl;
42. cout<<"I = "<<S2<<endl;
43.
44.
45. }

# Анализ результатов тестирования

Используемая функция: y=sin x \* cos x + x^3 + 8x \* sqrt(x).
Отрезок интегрирования – [0 ; 15].

Точность E = 10^(-6)

Вывод программы:
Результат = 15446,7
Оценка Рунге = 9,536 \* 10^(-7)
Количество разбиений = 524288

Точность E = 1

Вывод программы:
Результат = 15449,7
Оценка Рунге = 0.25
Количество разбиений = 2

Действительный результат : 15445

# Вывод:

В процессе выполнения лабораторной работы был рассмотрен метод парабол (метод Симпсона). Была рассмотрена реализация этого метода с заданной точностью. Метод позволяет достаточно точно рассчитать значение интеграла