**Министерство науки и высшего образования РФ**

**ФГБОУ ВО**

**Московский авиационный институт**

**(национальный исследовательский университет)**

Институт №3.

Системы управления, информатика и электроэнергетика.

Кафедра 304.

**Отчет по лабораторной работе №11**

**по учебной дисциплине  
 «Программирование на языках высокого уровня»  
на тему  
«Изучение математических функций стандартной библиотеки math.h и стандартной библиотеки stdlib.h»**

Выполнили студенты группы М3О-224Б-18:

Мариненко Г. В.

Рогачев В.Н.

Приняли:

Доцент кафедры 304, к-т н. Новиков П.В.\_\_\_\_\_\_

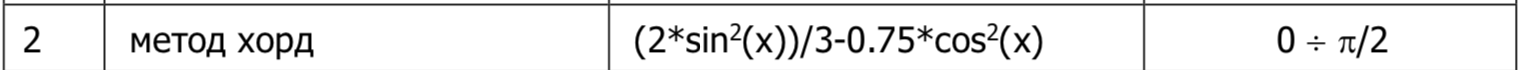
Доцент, к-т н. Чебатко М.И.\_\_\_\_\_\_

Москва 2019

**Задание:**

**1.**Разработать программу и подпрограмму (подпрограммы), вычисляющую корень уравнения с точностью ε = 0.001.

Метод приближения, функция и начальные координаты отрезка, содержащего корень, приведены в таблице.

**2.**Разработать программу и подпрограмму (подпрограммы), вычисляющую значение интеграла с точностью ε = 0.0005.

Интегрируемая функция и пределы интегрирования приведены в таблице, начальное число отрезков разбиения **n** = 10.


Структурная схема алгоритма программы:

Код программы:

#include <stdio.h>

#include "math.h"

#include "stdlib.h"

double ChordMethod(double \*function);

double Function1(double x); //Функция задания 1

double Function2(double x);

double Sum();

const double Eps1 = 0.001;

const double Eps2 = 0.0005;

int SplitLine = 10; //Число отрезков разбиения

int main()

{

double Task1 = ChordMethod(Function1);

printf("TASK 1: \n");

printf("Value of x = %lf \n", Task1);

printf("Value of F(x) = %lf \n", Function1(Task1));

double Task2 = Sum();

printf("TASK 2: \n");

printf("Value of Integral = %lf \n", Task2);

}

double Function1(double x)

{

return (2\*sin(x)\*sin(x))/3-0.75\*cos(x)\*cos(x);

}

double Function2(double x)

{

return sqrt(tan(x));

}

double ChordMethod(double \*function)

{

double x;

double A = 0;

double Pi = atan(1)\*4;

double B = Pi/2;

double Fa;

double Fb;

double Fx;

while(1)

{

Fa = Function1(A);

Fb = Function1(B);

x = (A\*Fb - B\*Fa)/(Fb-Fa);

Fx = Function1(x);

if (Fx \* Fa < 0)

{

B = x;

}

else

{

A = x;

}

if (fabs(Fx) < Eps1) break;

}

return x;

}

double Sum()

{

int i = 0;

double A = 0.0;

double Pi = atan(1)\*4;

double B = Pi/6;

double H = (B - A)/SplitLine;

double S = 0;

double NewS = 0;

for(i; i < (SplitLine - 1); i++)

{

S += Function2(A + H\*i + H/2);

}

S \*= H;

SplitLine++;

while (1)

{

H = (B - A)/SplitLine;

for(i; i < SplitLine - 1; i++)

{

NewS += Function2(A + H\*i + H/2);

}

NewS \*= H;

if(fabs((NewS - S)) < Eps2)

{

break;

}

else

{

SplitLine++;

S = NewS;

NewS = 0;

}

}

//printf("SplitLine = %d\n", SplitLine);

//printf("%lf\n", S);

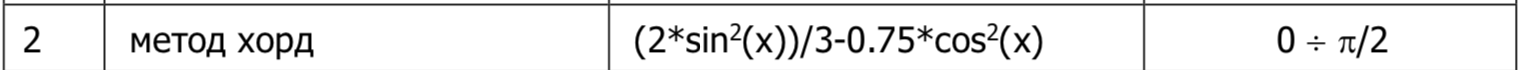
return NewS;

}

**Пример работы программы:**

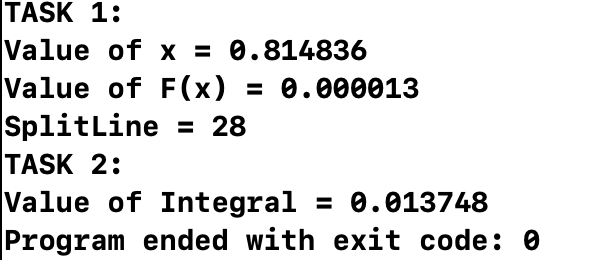
Условия входных данных:

**1.**

**2.**


****

**Вывод:**

Разработку программы считаем законченной на то основании, что:

1. На основе учебной лабораторной работы были закреплены знания по работе со строками, изучены новые функции.
2. Программы имеет достаточный набор тестов для корректной работы