Министерство образования и науки РФ

Санкт-Петербургский политехнический университет Петра Великого Институт компьютерных наук и технологий

Высшая школа программной инженерии

ОТЧЕТ

по лабораторной работе №1

Дисциплина «Вычислительная математика»

Выполнил

студентка гр. 3530904/90002 Мэн Цзянин

Руководитель С.П. Воскобойников

Санкт-Петербург

2021

# Задание:

# 

# Код решения задачи(c++)

#include <iostream>

#include <iomanip>

#include <cmath>

#include <fstream>

#include "SPLINES.H"

#include "quanc8.h"

using namespace std;

double point;

double h = 0.1;

double x\_beg = 2.0;

double x\_end = 3.0;

int count\_points = ((x\_end - x\_beg) / h) + 1;

double x\_beg\_compare = 2.0;

double posnR;

int N = 10;

double func(double x)

{

return ((1 - cos(point)) / point);

}

double omega(double x, int k)

{

double ans = 1; double xk = x\_beg;

for (int i = 0; i < count\_points; i++)

{

if (i != k)

ans \*= x - xk; xk += h;

}

return ans;

}

double Lagrange(double x, double\* f)

{

double ans = 0; double xk = x\_beg;

for (int i = 0; i < N; i++)

{

ans += (omega(x, i) / omega(xk, i)) \* f[i];

xk += h;

}

return ans;

}

int main()

{

double x[11], f[11];

x[0] = x\_beg;

ofstream s, l, q, in;

s.open("s.txt");

l.open("l.txt");

q.open("q.txt");

in.open("in.txt");

double ans1 = 0, ans2 = 0, errest;

int flag;

int col;

cout << "Integral (quanc8)" << endl;

for (int i = 0; i < count\_points; i++)

{

point = x[i];

quanc8(func, 0, x\_beg, 0.0001, 0.0001, &ans1, &errest, &col, &posnR, &flag);

cout << x[i] << " int=" << ans1 << " errest=" << errest << " " << " col="

<< col << " flag=" << flag << endl;

in << ans1 << endl;

f[i] = ans1;

if (i < (count\_points - 1))

x[i + 1] = x[i] + h;

}

double b[20], c[20], d[20];

double x\_s[20], f\_s[20];

for (int i = 0; i < count\_points; i++)

{

x\_s[i + 1] = x[i];

f\_s[i + 1] = f[i];

}

spline(11, x\_s, f\_s, b, c, d);

double xj = 0;

cout << "\n\n Comparing Integral(quanc8) and Spline" << endl;

cout << setw(8) << " t\t\t| " << "\tquanc8\tSpline\tQ-S" << endl;

xj = x\_beg\_compare;

for (int j = 0; j <= N; j++)

{

point = xj;

quanc8(func, 0, x\_beg, 0.0001, 0.0001, &ans1, &errest, &col, &posnR, &flag);

ans2 = seval(count\_points, &xj, x\_s, f\_s, b, c, d);

cout << setw(8) << xj << " | " << ans1 << " " << ans2 << "\t" << (ans1 - ans2) << endl;

s << ans2 << endl; q << ans1 << endl;

xj += 0.1;

}

cout << " " << endl;

cout << " Compareing (quanc8) and Lagrange" << endl;

cout << setw(8) << " t\t\t| " << "\tquanc8\tLagrange\tQ-L" << endl;

xj = x\_beg\_compare;

for (int j = 0; j <= N; j++)

{

point = xj;

quanc8(func, 0, x\_beg, 0.0001, 0.0001, &ans1, &errest, &col, &posnR, &flag);

ans2 = Lagrange(xj, f);

cout << setw(8) << xj << " | " << ans1 << " " << ans2 << "\t\t" << (ans1 - ans2) << endl;

l << ans2 << endl;

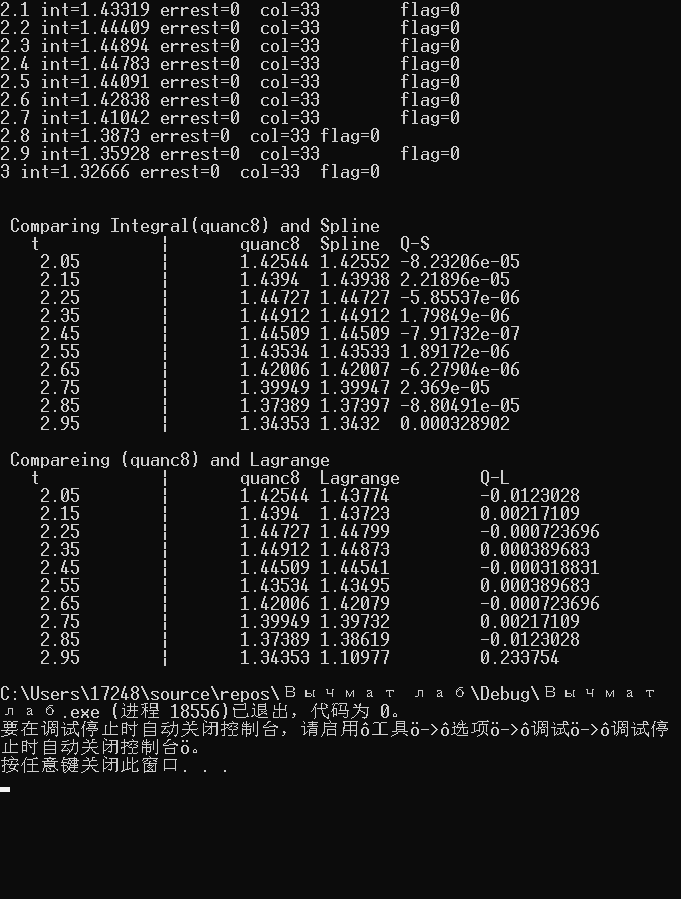
xj += 0.1;

}

return 0;

}

# Результат работы программы при заданных начальных данных



# Вывод

Результат данной работы совпадает с результатом подстановки от 2,05 до 2,95 с шагом 0,1. Более точнее значения получены при помощи использования spline функции.