ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ «САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ, МЕХАНИКИ И ОПТИКИ»

Факультет безопасности информационных технологий

Дисциплина:

«*Вычислительная математика*»

Отчёт по домашней работе №3

«Сплайн интерполяция»

Выполнил:

Назаров Максим Вячеславович

Группа:

N3248

Проверил:

д.т.н. Гришенцев Алексей Юрьевич

Санкт-Петербург 2022 г.

Сплайн интерполяция

Найти приближение функции заданной в равноотстоящих точках, т.е. функции, заданной в виде последовательности чисел, с помощью сплайна третьей степени.

Произвести анализ результатов. Предусмотреть возможность выбора размера последовательности преобразования без перекомпиляции программы. Размер последовательности N, где N = 5, 6, 7, … . Оценить вычислительную сложность.

Теория

Сплайн – функция, которая вместе с несколькими производными непрерывна на всем заданном отрезке [a, b], а на каждом частичном отрезке [x_i, x_{i+1}] в отдельности является некоторым алгебраическим многочленом.

Степенью сплайна называется максимальная по всем частичным отрезкам степень многочленов, а дефектом сплайна - разность между степенью сплайна и порядком наивысшей непрерывной на [a, b] производной. Например, непрерывная ломанная является сплайном степени 1 с дефектом 1 (так как сама функция – непрерывна, а первая производная уже разрывна).

Теория

На практике наиболее часто используются кубические сплайны $S_3(x)$ - сплайны третьей степени с непрерывной, по крайней мере, первой производной. При этом величина $m_i = S'_3(xi)$, называется наклоном сплайна в точке x_i.

Разобьём отрезок $[a, b]$ на N равных отрезков $[x_i, x_{i+1}]$, где $x_i = a + ih$, i=0, 1, …, N-1, $x_N = b$, $h = (b - a)/N$.

Теория

- Если в узлах x_i, x_{i+1} заданы значения f_i, f_{i+1}, которые принимает кубический сплайн, то на частичном отрезке $[x_i, x_{i+1}]$ он принимает вид:

$$S_3(x) = \frac{(x_{i+1}-x)^2(2(x-x_i)+h)}{h^3} f_i + \frac{(x-x_i)^2(2(x_{i+1}-x)+h)}{h^3} f_{i+1} + \frac{(x_{i+1}-x)^2(x-x_i)}{h^3} m_i + \frac{(x_{i+1}-x)^2(-x_{i+1}+x)}{h^3} m_{i+1}$$

(1)

В самом деле, это легко проверить, рассчитав $S_3(x)$ и $S_3'(x)$ в точках x_i, x_{i+1}. Можно доказать, что если многочлен третьей степени принимает в точках x_i , x_{i+1} значения f_i , f_{i+1} и имеет в этих точках производные, соответственно, m_i, m_{i+1}, то он совпадает с многочленом (1).

Теория

Таким образом, для того, чтобы задать кубический сплайн на отрезке, необходимо задать значения f_i, m_i i=0,1…, N в N+1 в узле x_i.

Кубический сплайн, принимающий в узлах те же значения f_i , что и некоторая функция, называется интерполяционным и служит для аппроксимации функции f на отрезке $[a, b]$ вместе с несколькими производными.

Программа

```
#include <vector>
#include <array>
#include <stdio.h>
#include <cmath>
using namespace std;
typedef array<double,4> polynome;
int main() {
	double a, b;
	int n;
		vector<double> m = {};
		vector<double> x = {};
	vector<double> f = {};
	printf("Введите границы: ");
	scanf("%lf %lf",&a,&b);
	printf("Введите N: ");
	scanf("%d",&n);
	double h = (b-a)/n;
	for (int i = 0; i <= n; i++) {
		x.push_back(a+i*h);
		printf("f(%lf): ",x[i]);
		double tmp;
		scanf("%lf",&tmp);
		f.push_back(tmp);
	}
```

Программа

```
m.push_back((4*f[1]-f[2]-3*f[0])/(2*h));
for (int i = 1; i < n; i++)
        m.push_back((f[i+1]-f[i-1])/(2*h));
m.push_back((3*f[n]-f[n-2]-3*f[n-1])/(2*h));

for (int i = 0; i < n; i++) {
        polynome s = {
        (-2*x[i]*pow(x[i+1],2)+h*pow(x[i+1],2))/pow(h,3)*f[i]+
        (2*x[i+1]*pow(x[i],2)+h*pow(x[i],2))/pow(h,3)*f[i+1]+
        (x[i]*pow(x[i+1],2))/pow(h,2)*m[i]+
        (x[i+1]*pow(x[i],2))/pow(h,2)*m[i+1],


        (2*pow(x[i+1],2)+4*x[i]*x[i+1]-2*x[i+1]*h)/pow(h,3)*f[i]+
        (-4*x[i]*x[i+1]-2*x[i]*h-2*pow(x[i],2))/pow(h,3)*f[i+1]+
        (pow(x[i+1],2)+2*x[i]*x[i+1])/pow(h,2)*m[i]+
        (2*x[i]*x[i+1]+pow(x[i],2))/pow(h,2)*m[i+1],


        (-4*x[i+1]-2*x[i]+h)/pow(h,3)*f[i]+
        (4*x[i]+2*x[i+1]+h)/pow(h,3)*f[i+1]+
        (-2*x[i+1]-x[i])/pow(h,2)*m[i]+
        (-2*x[i]-x[i+1])/pow(h,2)*m[i+1]
```

Пример вывода

Введите границы: 0 10

Введите N: 3

f(0.000000): 1

f(3.333333): 2

f(6.666667): 5

f(10.000000): 6

x ∈ [0.000000, 3.333333], S3(x) = (0.000000)x^3 + (0.090000)x^2 + (0.000000)x + (1.000000)

x ∈ [3.333333, 6.666667], S3(x) = (-0.054000)x^3 + (0.810000)x^2 + (-3.000000)x + (5.000000)

x ∈ [6.666667, 10.000000], S3(x) = (0.013500)x^3 + (-0.405000)x^2 + (4.200000)x + (-9.000000)

Трудоемкость

Выводы

В ходе выполнения домашнего задания я написал программу на языке C++; для нахождения приближения функции с помощью сплайна третьей степени; произвел анализ результатов. В отчете предоставил текст задачи, код программы, а также вычислил трудоемкость алгоритма.

Источники

Спасибо за внимание!