**Министерство науки и высшего образования Российской Федерации**

федеральное государственное автономное образовательное учреждение

высшего образования

«Санкт-Петербургский политехнический университет Петра Великого»

(ФГАОУ ВО «СПбПУ»)

**Институт среднего профессионального образования**

**Практическая работа №7**

**«Определение корней алгебраических и**

**трансцендентных уравнений.»**

**по предмету «Численные методы»**

**Тема: «Вычисление интерполяционного многочлена по методу Лагранжа»**

**Специальность: 09.02.07**

**Курс: 32919/4**

**Выполнил:** Зажарская А.В.

**Проверил:**  Кудрявцева Е.В.

**Оценка:**

Санкт-Петербург

19.11.2023

Задание 1: решить систему уравнений методом Лагранжа.

Дано:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | 1 | 3 | 5 | 6 |
| F(x) | 4 | 3 | 2 | -3 |

Решение:

Код был написан на языке С++.

Точность (е) = 0,1.

#include <iostream>

#include <vector>

using namespace std;

// Функция для вычисления значения интерполяционного многочлена Лагранжа в точке x\_val

double lagrange(vector<double> x, vector<double> y, double x\_val) {

double result = 0.0; // Инициализация результата

for (int i = 0; i < x.size(); i++) { // Цикл по всем точкам из таблицы

double term = y[i]; // Инициализация текущего слагаемого

for (int j = 0; j < x.size(); j++) { // Цикл по всем точкам из таблицы

if (j != i) { // Если точка не равна текущей

// Вычисление текущего слагаемого

term \*= (x\_val - x[j]) / (x[i] - x[j]);

}

}

result += term; // Добавление текущего слагаемого к результату

}

return result; // Возврат результата

}

int main() {

// Инициализация таблицы точек

vector<double> x = {1, 3, 5, 6};

vector<double> y = {4, 3, 2, -3};

double x\_val = 2; // Значение x, для которого нужно найти значение интерполяционного многочлена (можно заменить на любое другое число)

double result = lagrange(x, y, x\_val); // Вычисление значения интерполяционного многочлена

// Вывод результата

cout << "Значение интерполяционного многочлена в точке " << x\_val << " равно " << result << endl;

return 0;

}



Рисунок 1 – вывод



Рисунок 2 – вывод



Рисунок 3 – вывод

Задание 2: решить систему уравнений методом интерполяции сплайном.

Дано:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| x | 1 | 3 | 5 | 6 |
| F(x) | 4 | 3 | 2 | -3 |

Решение:

Код был написан на языке С++

Точность (е) = 0,0001.

#include <iostream>

#include <vector>

#include <cmath>

using namespace std;

// Функция для вычисления сплайна

double spline(double x, const vector<double> & x\_v, const vector<double> & f\_v) {

int n = x\_v.size();

// Создание векторов для хранения промежуточных значений

vector<double> a(n), b(n - 1), d(n - 1), h(n - 1);

// Вычисление шага между узлами

for (int i = 0; i < n - 1; ++i) {

h[i] = x\_v[i + 1] - x\_v[i];

}

// Вычисление коэффициентов a

for (int i = 1; i < n - 1; ++i) {

a[i] = 3 \* (f\_v[i + 1] - f\_v[i]) / h[i] - 3 \* (f\_v[i] - f\_v[i - 1]) / h[i - 1];

}

// Вычисление коэффициентов b,d

for (int i = 0; i < n - 1; ++i) {

if (i == 0) {

b[i] = 2 \* h[i];

d[i] = a[i] / b[i];

} else if (i == n - 2) {

b[i] = 2 \* h[i - 1] + 4 \* h[i];

d[i] = (a[i] - a[i - 1] \* h[i - 1] / 2 - 3 \* (f\_v[i] - f\_v[i - 1]) \* h[i - 1] / h[i]) / b[i];

} else {

b[i] = 2 \* (h[i - 1] + h[i]);

d[i] = (a[i] - a[i - 1] \* h[i - 1] / 2 - 3 \* (f\_v[i] - f\_v[i - 1]) \* h[i - 1] / h[i]) / b[i];

}

}

// Обратный ход метода прогонки

for (int i = n - 3; i >= 0; --i) {

d[i] = (a[i] - h[i] \* d[i + 1]) / (2 \* h[i] + h[i + 1] \* d[i + 1]);

}

// Определение интервала, в котором находится x

int i = 0;

while (x > x\_v[i + 1]) {

++i;

}

// Вычисление разности между x и узлом

double dx = x - x\_v[i];

// Вычисление значения сплайна в точке x

return f\_v[i] + d[i] \* dx + (3 \* (f\_v[i + 1] - f\_v[i]) / h[i] - 2 \* d[i] - d[i + 1]) \* dx \* dx / h[i] + (d[i] + d[i + 1] - 2 \* (f\_v[i + 1] - f\_v[i]) / h[i]) \* dx \* dx \* dx / (h[i] \* h[i]);

}

// Задание точки, в которой нужно вычислить значение функции double x = ...

int main() {

vector<double> x\_v = {1, 3, 5, 6};

vector<double> f\_v = {4, 3, 2, -3};

double x = 3;

cout << "Значение функции в точке " << x << " равно " << spline(x, x\_v, f\_v) << endl;

return 0;

}



Рисунок 4 – вывод



Рисунок 5 – вывод



Рисунок 6 – вывод