**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: 3**

**Выполнил:** Кабачарова А.Ю.

**Проверил:** Кудрявцева Е.В.

**Оценка:**

Санкт-Петербург

23.11.2023

**Задание:** решить систему уравнений методом Лагранжа.

**Дано:**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| x | 1 | 2 | 3 | 4 | 5 |
| F(x) | 48 | 24 | 8 | 12 | 24 |

**Решение:**

Код написан на языке программирования С++

#include <iostream>

#include <vector>

using namespace std;

double interpolateLagrange(const vector<double>& x, const vector<double>& f, double xi) {

double result = 0.0;

for (size\_t i = 0; i < x.size(); i++) {

double term = f[i];

for (size\_t j = 0; j < x.size(); j++) {

if (j != i) {

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

}

}

result += term;

}

return result;

}

int main() {

vector<double> x = {1, 2, 3, 4, 5};

vector<double> f = {48, 24, 8, 12, 24};

double xi = 12; // Ваше значение xi для интерполяции

double interpolatedValue = interpolateLagrange(x, f, xi);

cout << "При x = " << xi << ", F(x) = " << interpolatedValue << endl;

return 0;

}

Изображение выглядит как текст, снимок экрана, Мультимедийное программное обеспечение, программное обеспечение

Автоматически созданное описание   
Рисунок 1- вывод

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание   
Рисунок 2 - вывод

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Автоматически созданное описание  
Рисунок 3 - вывод