**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ**

Федеральное государственное бюджетное образовательное учреждение   
высшего образования

**«Сибирский государственный университет науки и технологий   
имени академика М.Ф. Решетнева»**

Институт информатики и телекоммуникаций

Кафедра информатики и вычислительной техники

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ №4**

Вычислительная математика

|  |
| --- |
| Интерполяция функций |

Руководитель К.А.Кириллов

подпись, дата инициалы, фамилия

Обучающийся БПИ20-02, 201219047 Р.А.Сухачев

номер группы, зачетной книжки подпись, дата инициалы, фамилия

Красноярск 2022 г.

# ЦЕЛЬ РАБОТЫ

Изучить интерполяцию функций.

# постановка задачи

Вариант 27:

(1)

1. Вычислить значения:

а) *fj*=*f*(*tj*) заданной функции (1) в равноотстоящих узлах

*tj*=(*j*-1)*h* (*j*=1,2,…,*n*+1), *h*=1/*n*

отрезка [0,1];

б) разделенных разностей первого порядка

= (2)

в) разделенных разностей второго порядка

= (3)

. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .

г) разделенных разностей n-го порядка

= (4)

данной функции.

1. По вычисленным в п. 1 значениям, используя интерполяционный многочлен Ньютона степени n

* …,
* найти значения *Nn*(*t*) в точках (узлах) с полуцелыми индексами

1. Найти погрешность интерполирования в этих узлах

максимальную погрешность

и среднюю квадратическую погрешность

.

Взять значение n=10.

# ХОД РАБОТЫ

**Общие теоретические сведения:**

1. Для вычисления значений были использованы формулы (1)-(4).

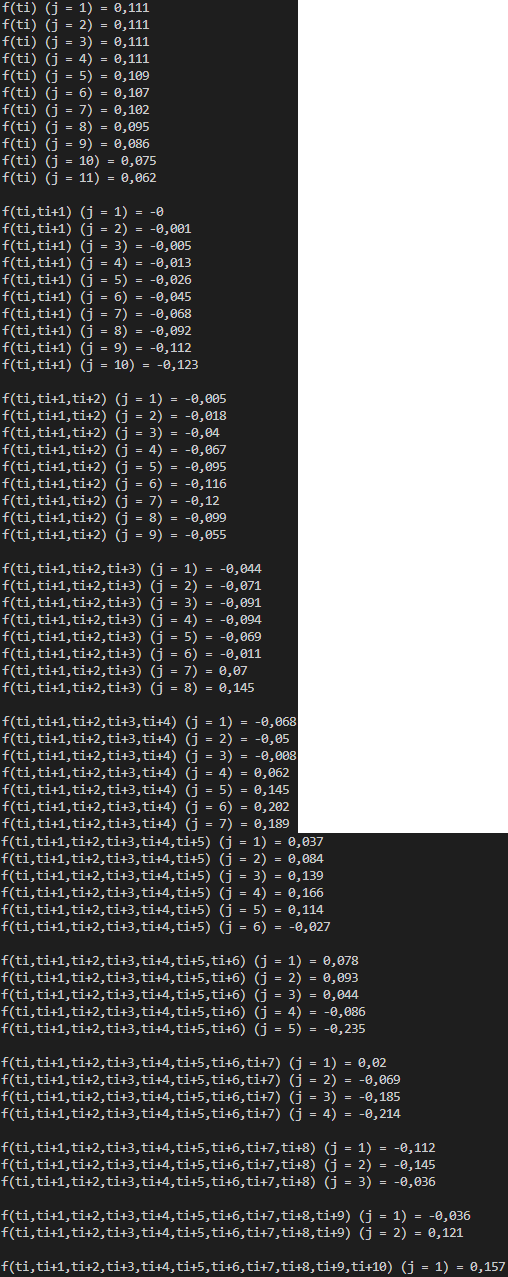


Рисунок 1 – Значения разделённых разностей.

1. Формула для интерполяционного многочлена Ньютона *N*10(*t*) 10-  
   й степени:

+

С подставленными значениями:

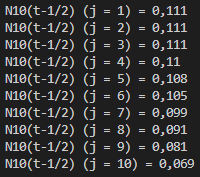


Рисунок 2 – значения *N*10(*t*) в точках с полуцелыми индексами.

1. Вычисления погрешностей:

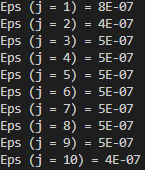


Рисунок 3 – Погрешность   
интерполирования в точках.

4

Рисунок 4 – Максимальная погрешность.

5

Рисунок 5 – Средняя квадратическая погрешность.

**Текст программы**

Содержимое файла Program.cs:

using System;

namespace lab3VAR27

{

    class Program

    {

        const double h = 0.1;

        static double f(double t)

        {

            return Math.Pow((2+(t\*t)), 1/2)\*Math.Pow((3+(t\*t\*t\*t)), -2);

        }

        static double t(int j)

        {

            return (j-1)\*h;

        }

        static double f1(int j)

        {

            return (f(t(j+1))-f(t(j)))/h;

        }

        static double f2(int j)

        {

            return (f1(j+1)-f1(j))/(2\*h);

        }

        static double f3(int j)

        {

            return (f2(j+1)-f2(j))/(3\*h);

        }

        static double f4(int j)

        {

            return (f3(j+1)-f3(j))/(4\*h);

        }

        static double f5(int j)

        {

            return (f4(j+1)-f4(j))/(5\*h);

        }

        static double f6(int j)

        {

            return (f5(j+1)-f5(j))/(6\*h);

        }

        static double f7(int j)

        {

            return (f6(j+1)-f6(j))/(7\*h);

        }

        static double f8(int j)

        {

            return (f7(j+1)-f7(j))/(8\*h);

        }

        static double f9(int j)

        {

            return (f8(j+1)-f8(j))/(9\*h);

        }

        static double f10(int j)

        {

            return (f9(j+1)-f9(j))/(10\*h);

        }

        static double t\_j(int j)

        {

            return (j-0.5)\*h;

        }

        static double ff1(int j)

        {

            return (t\_j(j)-t(1))\*f1(1);

        }

        static double ff2(int j)

        {

            return f2(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2));

        }

        static double ff3(int j)

        {

            return f3(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3));

        }

        static double ff4(int j)

        {

            return f4(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4));

        }

        static double ff5(int j)

        {

            return f5(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4))\*(t\_j(j)-t(5));

        }

        static double ff6(int j)

        {

            return f6(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4))\*(t\_j(j)-t(5))\*(t\_j(j)-t(6));

        }

        static double ff7(int j)

        {

            return f7(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4))\*(t\_j(j)-t(5))\*(t\_j(j)-t(6))\*(t\_j(j)-t(7));

        }

        static double ff8(int j)

        {

            return f8(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4))\*(t\_j(j)-t(5))\*(t\_j(j)-t(6))\*(t\_j(j)-t(7))\*(t\_j(j)-t(8));

        }

        static double ff9(int j)

        {

            return f9(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4))\*(t\_j(j)-t(5))\*(t\_j(j)-t(6))\*(t\_j(j)-t(7))\*(t\_j(j)-t(8))\*(t\_j(j)-t(9));

        }

        static double ff10(int j)

        {

            return f10(1)\*(t\_j(j)-t(1))\*(t\_j(j)-t(2))\*(t\_j(j)-t(3))\*(t\_j(j)-t(4))\*(t\_j(j)-t(5))\*(t\_j(j)-t(6))\*(t\_j(j)-t(7))\*(t\_j(j)-t(8))\*(t\_j(j)-t(9))\*(t\_j(j)-t(10));

        }

        static double N(int j)

        {

            return f(t\_j(1)) + ff1(j) + ff2(j) + ff3(j) + ff4(j) + ff5(j) + ff6(j) + ff7(j) + ff8(j) + ff9(j) + ff10(j);

        }

        static double eps(int j)

        {

            return Math.Abs(f(t\_j(j))-N(j));

        }

        static void Main(string[] args)

        {

            Console.Clear();

            Console.OutputEncoding = System.Text.Encoding.UTF8;

            int n = 10;

            Console.WriteLine("\nStart:\n");

            for (int j = 1; j <= n+1; j++)

            {

                Console.WriteLine($"f(ti) (j = {j}) = {Math.Round(f(t(j)), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n; j++)

            {

                Console.WriteLine($"f(ti,ti+1) (j = {j}) = {Math.Round(f1(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-1; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2) (j = {j}) = {Math.Round(f2(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-2; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3) (j = {j}) = {Math.Round(f3(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-3; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4) (j = {j}) = {Math.Round(f4(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-4; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4,ti+5) (j = {j}) = {Math.Round(f5(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-5; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4,ti+5,ti+6) (j = {j}) = {Math.Round(f6(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-6; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4,ti+5,ti+6,ti+7) (j = {j}) = {Math.Round(f7(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-7; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4,ti+5,ti+6,ti+7,ti+8) (j = {j}) = {Math.Round(f8(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-8; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4,ti+5,ti+6,ti+7,ti+8,ti+9) (j = {j}) = {Math.Round(f9(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n-9; j++)

            {

                Console.WriteLine($"f(ti,ti+1,ti+2,ti+3,ti+4,ti+5,ti+6,ti+7,ti+8,ti+9,ti+10) (j = {j}) = {Math.Round(f10(j), 3)}");

            }

            Console.WriteLine($"\nN10(t) = {Math.Round(f(t(1)), 3)} + (t-{Math.Round(t(1), 3)})({Math.Round(f1(1), 3)}) + ({Math.Round(f2(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)}) + ({Math.Round(f3(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)}) + ({Math.Round(f4(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)}) + ({Math.Round(f5(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)})(t-{Math.Round(t(5), 3)}) + ({Math.Round(f6(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)})(t-{Math.Round(t(5), 3)})(t-{Math.Round(t(6), 3)}) + ({Math.Round(f7(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)})(t-{Math.Round(t(5), 3)})(t-{Math.Round(t(6), 3)})(t-{Math.Round(t(7), 3)}) + ({Math.Round(f8(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)})(t-{Math.Round(t(5), 3)})(t-{Math.Round(t(6), 3)})(t-{Math.Round(t(7), 3)})(t-{Math.Round(t(8), 3)}) + ({Math.Round(f9(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)})(t-{Math.Round(t(5), 3)})(t-{Math.Round(t(6), 3)})(t-{Math.Round(t(7), 3)})(t-{Math.Round(t(8), 3)})(t-{Math.Round(t(9), 3)}) + ({Math.Round(f10(1), 3)})(t-{Math.Round(t(1), 3)})(t-{Math.Round(t(2), 3)})(t-{Math.Round(t(3), 3)})(t-{Math.Round(t(4), 3)})(t-{Math.Round(t(5), 3)})(t-{Math.Round(t(6), 3)})(t-{Math.Round(t(7), 3)})(t-{Math.Round(t(8), 3)})(t-{Math.Round(t(9), 3)})(t-{Math.Round(t(10), 3)})");

            Console.WriteLine();

            for (int j = 1; j <= n; j++)

            {

                Console.WriteLine($"N10(t-1/2) (j = {j}) = {Math.Round(N(j), 3)}");

            }

            Console.WriteLine();

            for (int j = 1; j <= n; j++)

            {

                Console.WriteLine($"Eps (j = {j}) = {Math.Round(eps(j), 7)}");

            }

            Console.WriteLine();

            var A = new List<Double>();

            for (int j = 1; j <= n; j++)

            {

                A.Add(eps(j));

            }

            Console.WriteLine($"Максимальная погрешность = {Math.Round(A.Max(), 7)}");

            for (int j = 1; j < A.Count(); j++)

            {

                A[j] = A[j]\*A[j];

            }

            Console.WriteLine($"\nСреднеквадратичная погрешность = {Math.Sqrt(A.Sum()\*1/n)}\nEnd.\n");

        }

    }

}

# ВЫВОДЫ

В ходе работы была изучена интерполяция функции.