**Санкт-Петербургский национальный исследовательский университет**

**информационных технологий, механики и оптики**

**Кафедра информатики и прикладной математики**

Вычислительная математика

Лабораторная работа №3

Метод Лагранжа

Выполнил: Гхази Даниэль

Группа P3218

Преподаватель: Исаев Илья Владимирович

г. Санкт-Петербург

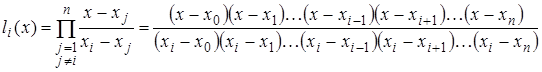
2017 г.

**1. Описание метода**

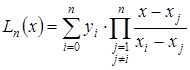
При глобальной интерполяции на всем интервале http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image025_0002.png строится единый многочлен. Одной из форм записи интерполяционного многочлена для глобальной интерполяции является многочлен Лагранжа:

http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image095.png

где http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image097.png– базисные многочлены степени *n*:



То есть многочлен Лагранжа:



Многочлен http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image097_0000.png удовлетворяет условию http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image103.png.  Это условие означает, что многочлен равен нулю при каждом http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image105.png кроме http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image016_0002.png, то есть http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image108.png – корни этого многочлена. Таким образом, степень многочлена http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image110.png равна *n* и при http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image112.png в сумме обращаются в нуль все слагаемые, кроме слагаемого с номером http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image114.png, равного http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image116.png.

Выражение (3.11) применимо как для равноотстоящих, так и для не равноотстоящих узлов. Погрешность интерполяции методом Лагранжа зависит от свойств функции http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image002_0001.png, от расположения узлов интерполяции и точки *x*. Полином Лагранжа имеет малую погрешность при небольших значениях *n* (*n*<20). При больших *n* погрешность начинает расти, что свидетельствует о том, что метод Лагранжа не сходится (т.е. его погрешность не убывает с ростом *n*).

Многочлен Лагранжа в явном виде содержит значения функций в узлах интерполяции, поэтому он удобен, когда значения функций меняются, а узлы интерполяции неизменны. Число арифметических операции, необходимых для построения многочлена Лагранжа, пропорционально http://aco.ifmo.ru/el_books/numerical_methods/lectures/glava3/glava3_clip_image118.png и является наименьшим для всех форм записи. К недостаткам этой формы записи можно отнести то, что с изменением числа узлов приходится все вычисление проводить заново.

Кусочно-линейная и кусочно-квадратичная локальные интерполяции являются частными случаями интерполяции многочленом Лагранжа.

**2. Блок-схема**

****

**3. Реализация метода**

class Lab3Lagr

{

public delegate double Function(double x);

public delegate double FunctionCoef(double x0, double x1);

static Function Func;

static FunctionCoef Coef;

public static List<Spot> Evaluate(double leftBorder, double rightBorder, double[,] inputTable, int size, double dx)

{

List<Spot> result = new List<Spot>();

Function f = LagrangesMethod(inputTable, size);

for (double x = leftBorder; x < rightBorder; x += dx)

{

result.Add(new Spot(x, f(x)));

}

return result;

}

public static Function LagrangesMethod(double[,] inputTable, int size)

{

Coef = (x1, x0) => 1 / (x1 - x0);

Func = x =>

{

double composition = 1;

double sum = 0;

double c = 1;

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

{

composition = 1;

c = 1;

double y = inputTable[1, i];

double xi = inputTable[0, i];

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

{

if (i != j)

{

double xj = inputTable[0, j];

composition \*= (x - xj);

c \*= Coef(xi, xj);

}

}

composition \*= y \* c;

sum += composition;

}

return sum;

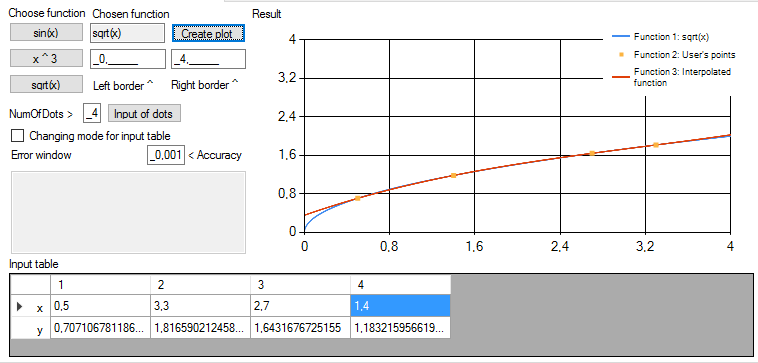
};

return Func;

}

}

**4. Пример работы программы**



**5. Вывод**

В результате выполнения данной лабораторной работы я ознакомился с методами интерполяции функций и пришел к выводу о том, что метод Лагранжа является лучшим среди изученных, так как узлы интерполяции, подаваемые ему на вход, не должны быть равноотстоящими и упорядоченными по величине, как необходимо в методе Ньютона. Полученная в результате функция имеет степень не превосходящую количество заданных пар чисел (х и у), в отличии от метода кубического сплайна.