МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра «Програмна інженерія та інформаційні технології управління»

Звіт з індивідуального розрахункового завдання №12

З предмету «Числові методи»

Виконав

Студент групи КН-36а

Рубан Ю.Д.

Перевірив:

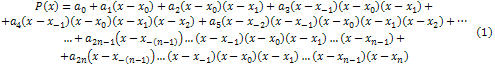
Гужва В.О.

Харків - 2017

Завдання: знайти інтерполяційний поліном Гаусса по точкам за другою формулою

Нехай маємо Інтерполяційна формула Гаусса рівновіддалених вузлів інтерполяції Інтерполяційна формула Гаусса, де Інтерполяційна формула Гаусса, і для деякої функції Інтерполяційна формула Гаусса відомо її значення в даних вузлах, тобто Інтерполяційна формула Гаусса. Задача полягає у побудові інтерполяційного полінома, степінь якого не перевищує Інтерполяційна формула Гаусса і значення якого у візлах інтерполяції співпадає з відомими значеннями функції (Інтерполяційна формула Гаусса).

Даний поліном будемо шукати в наступному вигляді:



Використовуючи узагальнену степінь числа, вираз (1) перепишемо у наступному вигляді:

Інтерполяційна формула Гаусса

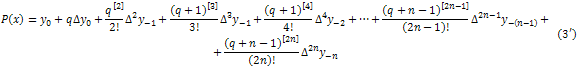
Застосовуючи для обчислення коефіцієнтів Інтерполяційна формула Гаусса той самий спосіб, що і при виводі [інтерполяційної формули Ньютона](http://www.mathros.net.ua/persha-interpoljacijna-formula-njutona-dlja-rivnoviddalenyh-vuzliv-interpoljacii.html" \o "Перша інтерполяційна формула Ньютона для рівновіддалених вузлів інтерполяції" \t "_blank), послідовно знаходимо:

Інтерполяційна формула Гаусса

Далі, ввівши змінну Інтерполяційна формула Гаусса і зробивши відповідну заміну в формулі (2) отримуємо першу інтерполяційну формулу Гаусса:

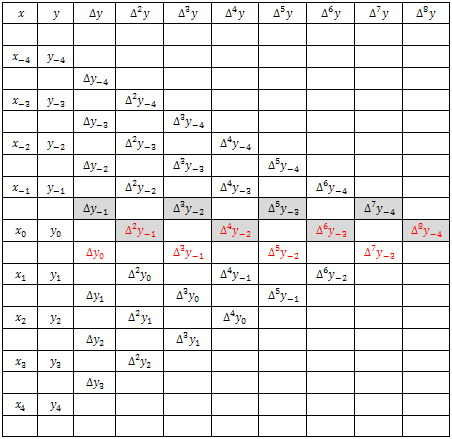
Інтерполяційна формула Гаусса

або в скороченій формі:



де Інтерполяційна формула Гаусса.

Перша інтерполяційна формула Гаусса використовує центральні скінченні різниці Інтерполяційна формула Гаусса (в таблиці утворюють нижню ламану люнію позначену червоним кольором).

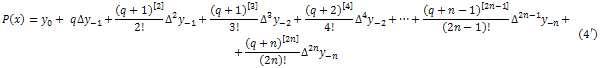


Таблиця центральних скінченних різниць

Аналогічним чином можна отримати другу інтерполяційну формулу Гаусса, яка використовує центральні різниці виду Інтерполяційна формула Гаусса(утворюють верхню ламану лінію позначену сірим кольором) і записується в наступному вигляді:

Інтерполяційна формула Гаусса

або в скороченій формі:



де Інтерполяційна формула Гаусса.

Зауваження: перша інтерполяційна формула Гаусса використовується для обчислення наближеного значення функції в тому випадку, коли Інтерполяційна формула Гаусса, тоді, як друга — коли Інтерполяційна формула Гаусса.

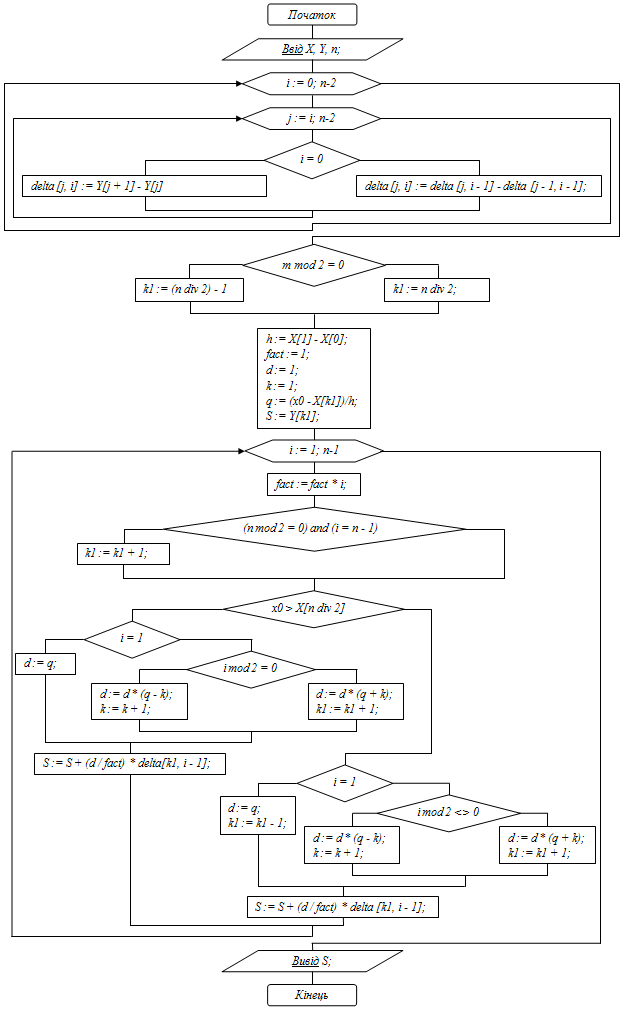


Рисунок 1 – Блок схема алгоритму інтерполяції методом Гаусса

Ручне рішення

|  |  |
| --- | --- |
| 7 | 2 |
| 10 | 4 |
| 13 | 14 |
| 16 | 20 |
| 19 | 10 |

2 10 6 -10

8 -4 -16

-12 -12

0

h = x(i+1) - x(i) = 3

q = (x - x0)/h = (x - 13)/3

f(x) = 14 + ((q + 0) \* 6/1!) + ((q + 1)(q + 0) \* -4/2!) + ((q + 1)(q + 0)(q - 1) \* -12/3!) + ((q + 2)(q + 1)(q + 0)(q - 1) \* 0/4!)

х = 11

f(11) = 7.03704

Фрагмент коду програми

#include"Gauss\_interpolation\_polynomal.h"

Gauss\_interpolation\_polynomal::Gauss\_interpolation\_polynomal(vector<vector<double>>matrix, int size)

{

deltaY.resize(size - 1);

int delta0 = size - 1;

for (int i = delta0, k = 0; k<size - 1; i--, k++)

{

deltaY[k].resize(i);

}

for (int i = 0; i < delta0; i++)

{

for (int j = 0; j < deltaY.size() - i; j++)

{

if (i == 0)

deltaY[i][j] = matrix[j + 1][1] - matrix[j][1];

else

deltaY[i][j] = deltaY[i - 1][j + 1] - deltaY[i - 1][j];

}

}

ma = matrix;

Algorithms::show(deltaY);

}

double Gauss\_interpolation\_polynomal::do\_algorithm(double x)

{

int begin\_index;

if (ma.size() % 2 == 0)

begin\_index = (int)floor(ma.size() / 2) - 1;

else

begin\_index = (int)floor(ma.size() / 2);

double y = ma[begin\_index][1];

int f = 1;

double h = ma[1][0] - ma[0][0];

double qi = (x - ma[begin\_index][0])/h;

int size = ma.size();

double temp = 0;

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

{

y += build\_qi(i, qi)\*deltaY[i][begin\_index - ceil((float)i / 2.)] /Algorithms::fact(i+1);

}

return y;

}

double Gauss\_interpolation\_polynomal::build\_qi(int index, double q)

{

double res = 1;

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

{

res \*= q + (int)floor((float)index / 2.) - i;

}

return res;

}

string Gauss\_interpolation\_polynomal::get\_polynomal()

{

if (polynomal != "")

return polynomal;

else

{

int begin\_index;

if (ma.size() % 2 == 0)

begin\_index = (int)floor(ma.size() / 2) - 1;

else

begin\_index = (int)floor(ma.size() / 2);

polynomal += "h = x(i+1) - x(i) = " + toString(ma[1][0] - ma[0][0]) + "\n";

polynomal += "q = (x - x0)/h = (x - " + toString(ma[begin\_index][0]) + ")/" + toString(ma[1][0] - ma[0][0]) + "\n";

polynomal += "f(x) = ";

polynomal += toString(ma[begin\_index][1]);

polynomal += " + ";

double y = ma[ma.size() - 1][1];

double h = ma[1][0] - ma[0][0];

int size = ma.size();

for (int i = 0, k = size - 1; i < size - 1; i++, k--)

{

polynomal += "(";

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

{

if(((int)ceil((float)i / 2.) - j)>=0)

polynomal += "(q + " + toString((int)ceil((float)i / 2.) - j)+")";

else

polynomal += "(q - " + toString(-((int)ceil((float)i / 2.) - j)) + ")";

}

polynomal += " \* " + toString(deltaY[i][begin\_index - ceil((float)i / 2.)]);

polynomal += "/" + toString(i + 1) + "!) + ";

}

polynomal.erase(polynomal.end() - 2);

return polynomal;

}

}

Результат виконання програми

n=5

x,y =

7 2

10 4

13 14

16 20

19 10

xi = 11

f(xi) = 7.03704

Висновок:

Результати програми і ручного рішення збігаються

***Список використаних джерел***

1) Гончаров В. Л., Теория интерполирования и приближения функций, 2 изд., М., 1954;

2) Самарский А.А. Гулин А.В. Численные методы 1989г.

3) Костомаров Д.П., Фаворский А.П. Вводные лекции по численным методам