Name Of The Experiment: Gauss’ Central Difference Formulae.

Theory:

Gauss’ Forward Formula: The formula is, therefore ,of the form

Yp=y0+G1∆y0+G2∆2y-1+G3∆3y-1+G4∆4y-2+…………..

Where G1,G2,…… have to be determined . The yp on the left side can be expressed in terms of y0, ∆y0 and higher order differences of y0, as follows:

Table 1: Gauss’ Forward Formula

X Y ∆ ∆2  ∆3 ∆4  ∆5  ∆6

X­-3  y-3

∆y-3

X­-2  y-2 ∆2y-3

∆y-2  ∆3y-3

X­-1  y-1 ∆2y-2 ∆4y-3

∆y-1 ∆3y-2 ∆5y-3

X­0  y0 ∆2y-1 ∆4y-2 ∆6y-3

∆y0 ∆3y-1 ∆5y-2

X­1  y1 ∆2y0 ∆4y-1

∆y1 ∆3y0

X­2  y2 ∆2y1

∆y2

X­3  y3

G1=p,

G2=,

G3=,

G4=.

Gauss’ Backward Formula: The formula can therefore be assumed to be of the form

Yp=y0+G’1∆y-1+G’2∆2y-1+G’3∆3y-2+G’4∆4y-2+…………..

Table 2: Gauss’ Backward Formula

X Y ∆ ∆2  ∆3 ∆4  ∆5  ∆6

X­-3  y-3

∆y-3

X­-2  y-2 ∆2y-3

∆y-2  ∆3y-3

X­-1  y-1 ∆2y-2 ∆4y-3

∆y-1 ∆3y-2 ∆5y-3

X­0  y0 ∆2y-1 ∆4y-2 ∆6y-3

∆y0 ∆3y-1 ∆5y-2

X­1  y1 ∆2y0 ∆4y-1

∆y1 ∆3y0

X­2  y2 ∆2y1

∆y2

X­3  y3

Where,

G’1=p,

G’2=,

G’3=,

G’4=.

The source code for Gauss Backward and Forward Formula is given below:

#include<bits/stdc++.h>

using namespace std;

int main()

{

double x[1000],y[1000],del[1000],del2[1000],del3[1000],del4[1000],del5[1000],x1,p,h;

int i,j,k,n,m,ch;

cout<<"Enter the number of points:";

cin>>n;

cout<<"Enter the data point:"<<endl;

cout<<"x y=f(x)"<<endl;

for(i=0;i<n;i++) {

cin>>x[i]>>y[i]; }

while(1){

cout<<"1.Forward interpolation\t2.Backward interpolation\t0.Exit"<<endl;

cout<<"Enter your choice:";

cin>>ch;

if(ch==1){

cout<<"Enter the desired point:";

cin>>x1;

h=(x[1]-x[0]);

p=(x1-x[0])/h;

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

del[i]=y[i+1]-y[i];

cout<<"del["<<i<<"]="<<del[i]<<" "; }

for(i=0; i<n-2; i++) {

del2[i]=del[i+1]-del[i];

cout<<"del2["<<i<<"]="<<del2[i]<<" "; }

for(i=0; i<n-3; i++) {

del3[i]=del2[i+1]-del2[i];

cout<<"del3["<<i<<"]="<<del3[i]<<" "; }

for(i=0; i<n-4; i++){

del4[i]=del3[i+1]-del3[i];

cout<<"del4["<<i<<"]="<<del4[i]<<" "; }

double a;

a= y[0]+p\*del[0]+((p\*(p-1))\*del2[0])/2+((p\*(p-1)\*(p-2))\*del3[0])/6+((p\*(p-1)\*(p-2)\*(p-3))\*del4[0])/24;

cout<<"Required value is "<<a<<endl;}

else if(ch==2){

cout<<"Enter the desired point:";

cin>>x1;

h=(x[1]-x[0]);

p=(x1-x[n-1])/h;

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

del[i]=y[i+1]-y[i];

cout<<"del["<<i<<"]="<<del[i]<<" "; }

for(i=0; i<n-2; i++) {

del2[i]=del[i+1]-del[i];

cout<<"del2["<<i<<"]="<<del2[i]<<" "; }

for(i=0; i<n-3; i++) {

del3[i]=del2[i+1]-del2[i];

cout<<"del3["<<i<<"]="<<del3[i]<<" ";}

for(i=0; i<n-4; i++) {

del4[i]=del3[i+1]-del3[i];

cout<<"del4["<<i<<"]="<<del4[i]<<" "; }

double a1;

a1=y[n-1]+p\*del[n-2]+((p\*(p+1))\*del2[n-3])/2+((p\*(p+1)\*(p+2))\*del3[n-4])/6+((p\*(p+1)\*(p+2)\*(p+3))\*del4[n-5])/24;

cout<<"Required value is "<<a1<<endl; }

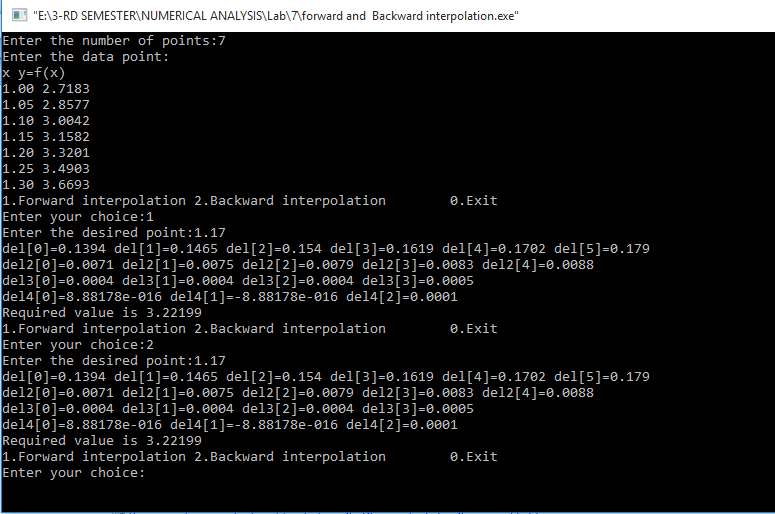
else if(ch==0) {

exit(1);}

}

return 0;

}



Discussion:The Gauss’ central difference formula is most suited for interpolation near the middle of a tabulated set .It is divided as Gauss’ forward formula and backward formula.The difference table of these two formula is different in the way of expressing .