**1. Write a program to find the value of f(x) at a point using Backward Interpolation difference table.**

**Algorithm:**

1. Enter the number of data points n and functional value f(x)
2. Input xp value at which interpolation is required.
3. Calculate n backward differences .
4. Calculate n degree Interpolation Polynomial as,

Pn(x) = f(xn) +

Where,

h = x1 – x0 and

1. Stop

**Source Code:**

//Backward differences

#include<stdio.h>

#include<conio.h>

#include<math.h>

#define MAX 10

long int fact(int n)

{

long int p=1;

int i;

for(i=1;i<=n;i++)

p\*=i;

return p;

}

int main()

{

int i,j,n;

float sum=0,pro,f[MAX],x[MAX],d[MAX][MAX],xp,h;

printf("Enter the no. of data points: ");

scanf("%d",&n);

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

{

printf("Enter the value of x[%d]: ",i); scanf("%f",&x[i]);

printf("Enter the value of f[%d]: ",i); scanf("%f",&f[i]);

}

printf("Enter the point x: "); scanf("%f",&xp);

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

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

d[i][0]=f[i];

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

for(i=0;i<n-j;i++)

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

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

printf("x(i)\t y(i)\t y1(i)\t y2(i)\t y3(i)\t y4(i)");

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

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

{

printf("\n%.3f",x[i]);

for(j=0;j<n-i;j++)

{

printf(" \t%.3f",d[i][j]);

}

printf("\n");

}

sum=f[n-1];

for(i=1;i<n;i++)

{ pro=1.0;

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

pro\*=(xp-x[n-j-1]);

sum+=(d[n-i-1][i]\*pro/(pow(h,i)\*fact(i)));

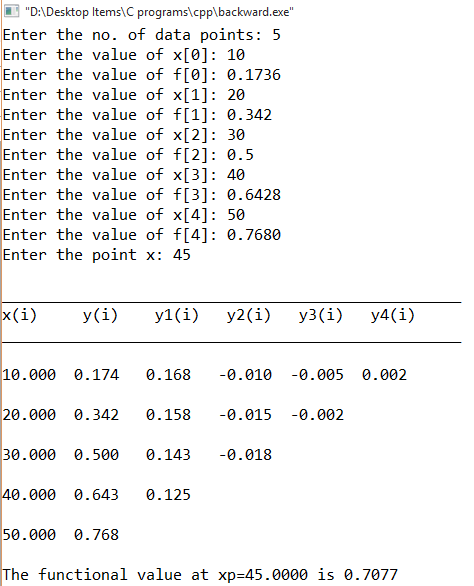
}

printf("\nThe functional value at xp=%.4f is %.4f",xp,sum);

return 0;

}

**Output**:

****

**2. Write a program to find the value of f(x) at a point using Forward Interpolation difference table.**

**Algorithm:**

1. Enter the number of data points n and functional value f(x)
2. Input xp value at which interpolation is required.
3. Calculate n forward differences .
4. Calculate n degree Interpolation Polynomial as,

Pn(x) = f(x0) +

Where,

h = x1 – x0 and

1. Stop

**Source Code:**

//Forward Difference

#include<stdio.h>

#include<conio.h>

#include<math.h>

#define MAX 10

long int fact(int n)

{

long int p=1;

int i;

for(i=1;i<=n;i++)

p\*=i;

return p;

}

int main()

{

int i,j,n;

float sum=0,pro,f[MAX],x[MAX],d[MAX][MAX],xp,h;

printf("Enter the no. of data points: ");

scanf("%d",&n);

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

{

printf("Enter the value of x[%d]: ",i); scanf("%f",&x[i]);

printf("Enter the value of f[%d]: ",i); scanf("%f",&f[i]);

}

printf("Enter the point x: "); scanf("%f",&xp);

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

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

d[i][0]=f[i];

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

for(i=0;i<n-j;i++)

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

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

printf("x(i)\t y(i)\t y1(i)\t y2(i)\t y3(i)\t y4(i)");

printf("\n\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

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

{

printf("\n%.3f",x[i]);

for(j=0;j<n-i;j++)

{

printf("\t%.3f",d[i][j]);

}

printf("\n");

}

sum=f[0];

for(i=1;i<n;i++)

{ pro=1.0;

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

pro\*=(xp-x[j]);

sum+=(d[0][i]\*pro/(pow(h,i)\*fact(i)));

}

printf("\nThe functional value at xp=%.4f is %.4f",xp,sum);

return 0;

}

**Output:**

