**Experiment – 12**

**Aim:** To interpolate the polynomial by Lagrange’s formula & to find the interpolated value.

**Software Used:** VS Code

**Theory:**

Let y = f(x) be a function, which takes the values (,) (, ),…..,(,)then the function f(x) can be represented by the polynomial as,

This is known as Lagrange’s interpolation formula.

**Algorithm:**

1. Start

2. Read number of data (n)

3. Read data Xi and Yi for i=1 ton n

4. Read value of independent variables say whose corresponding value of dependent say is to be determined.

5. Initialize: = 0

6. For i = 1 to n Set p = 1 For j =1 to n If i ≠ j then

Calculate p = p \* ( - )/( - )

End If

Next j Calculate = + p \* Yi

Next i

6. Display value of as interpolated value.

7. Stop

**Source Code:**

#include<iostream>

#include<conio.h>

using namespace std;

int main()

{

float x[100], y[100], xp, yp=0, p;

int i,j,n;

cout<<"\t Hritik\n\t\t26-IT-19";

cout<<"\n Interpolation by Lagrange's Formula\n";

/\* Input Section \*/

cout<<"Enter number of data: ";

cin>>n;

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

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

{

cout<<"x["<< i<<"] = ";

cin>>x[i];

cout<<"y["<< i<<"] = ";

cin>>y[i];

}

cout<<"Enter interpolation point: ";

cin>>xp;

/\* Implementing Lagrange Interpolation \*/

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

{

p=1;

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

{

if(i!=j)

{

p = p\* (xp - x[j])/(x[i] - x[j]);

}

}

yp = yp + p \* y[i];

}

cout<< endl<<"Interpolated value at "<< xp<< " is "<< yp;

return 0;

}

**Output:**

