ABHISHEK KUMAR

ROLL NO.- 20222756

PRACTICAL - 8

NEWTON'S INTERPOLATING POLYNOMIAL

⊙Computing Divided Difference

In[29]:=
```
NDD[x0_, f0_, startindex_, endindex_] :=
  Module[{x = x0, f = f0, i = startindex, j = endindex, answer},
   If[i == j, Return[f[[i]]], answer =
     (NDD[x, f, i + 1, j] - NDD[x, f, i, j - 1]) / (x[[j]] - x[[i]]);
    Return[answer]];];
```

In[30]:=
```
Ques - 1

x = {0, 1, 3};
f = {1, 3, 55};
NDD[x, f, 1, 2]
```

Out[30]= -1 + Ques

Out[33]= 2

In[34]:=
```
x = {0, 1, 3};
f = {1, 3, 55};
NDD[x, f, 2, 3]
```

Out[36]= 26

In[37]:=
```
NDD[x, f, 1, 3]
```

Out[37]= 8

In[38]:=
```
Ques - 2

x = {-1, 0, 1, 2};
f = {5, 1, 1, 11};
NDD[x, f, 1, 2]
```

Out[38]= $-2 + \text{Ques}$

Out[41]= -4

In[42]:=
```
NDD[x, f, 2, 3]
```

Out[42]= 0

⊙Computing Polynomial

In[72]:=
```
NDD[x0_, f0_] :=
 Module[{x1 = x0, f = f0, n, newtonPolynomial, k, j},
  n = Length[x1];
  newtonPolynomial[y_] = 0;
  For[i = 1, i ≤ n, i++, prod[y_] = 1;
   For[k = 1, k ≤ i - 1,
    k++, prod[y_] = prod[y] * (y - x1[[k]])];
   newtonPolynomial[y_] =
    newtonPolynomial[y] + NDD[x1, f, 1, i] * prod[y]];
  Return[newtonPolynomial[y]];];

nodes = {0, 1, 3};
values = {1, 3, 55};
NDD[nodes, values]
```

Out[75]= $1 + 2\,y + 8\,(-1 + y)\,y$

In[77]:=
```
Simplify[%]
```

Out[77]= $1 - 6\,y + 8\,y^2$

In[76]:=
```
NDD[{0, 1, 3}, {1, 3, 55}]
```

Out[76]= $1 + 2\,y + 8\,(-1 + y)\,y$