Lagrange Interpolation - Numerical Modelling (ESN-421)

Name: Prahlada V Mittal Enr : 20411023 Branch: GPT 4th Year

In [1]:

```
import numpy as np
import matplotlib.pyplot as plt
from scipy.interpolate import interp1d
```

In [5]:

```
# Function for performing lagrange interpolation

def lagrange_interp(xi, yi, x):
    n = len(xi)                     # Number of data points
    P_x = np.zeros_like(x)          # Initialized interpolated values

    for k in range(len(x)):
        weight = np.ones(n)         # Initialize weights for Lagrange interpolatio

        # Calculate weights using Lagrange formula
        for i in range(n):
            indices = []           # Initialize indices for excluding the curr

            for j in range(n):
                if j != i:         # Exclude the current index i
                    indices.append(j)   # Exclude the index i (j!=i condition)

            num = x[k] - xi[indices]            # (x - x_j)
            den = xi[i] - xi[indices]         # (x_i - x_j)
            weight[i] = np.prod(num / den)

        # Calculate the interpolated value at x[k]
        P_x[k] = np.sum(weight * yi)

    return P_x
```

In [6]:

```
# Example data (Testing)

n = 10
xi = np.linspace(-1, 1, n)
x = np.linspace(-1, 1, 1000)
yi = np.array([3, 2, 2, 1, 0, -3, -4, 1, 2, 5])  #given


# Lagrange interpolation
lagrange_y = lagrange_interp(xi, yi, x)
```

In [7]:
```
# Linear interpolation
linear_interp = interp1d(xi, yi, kind='linear')
linear_y = linear_interp(x)
```

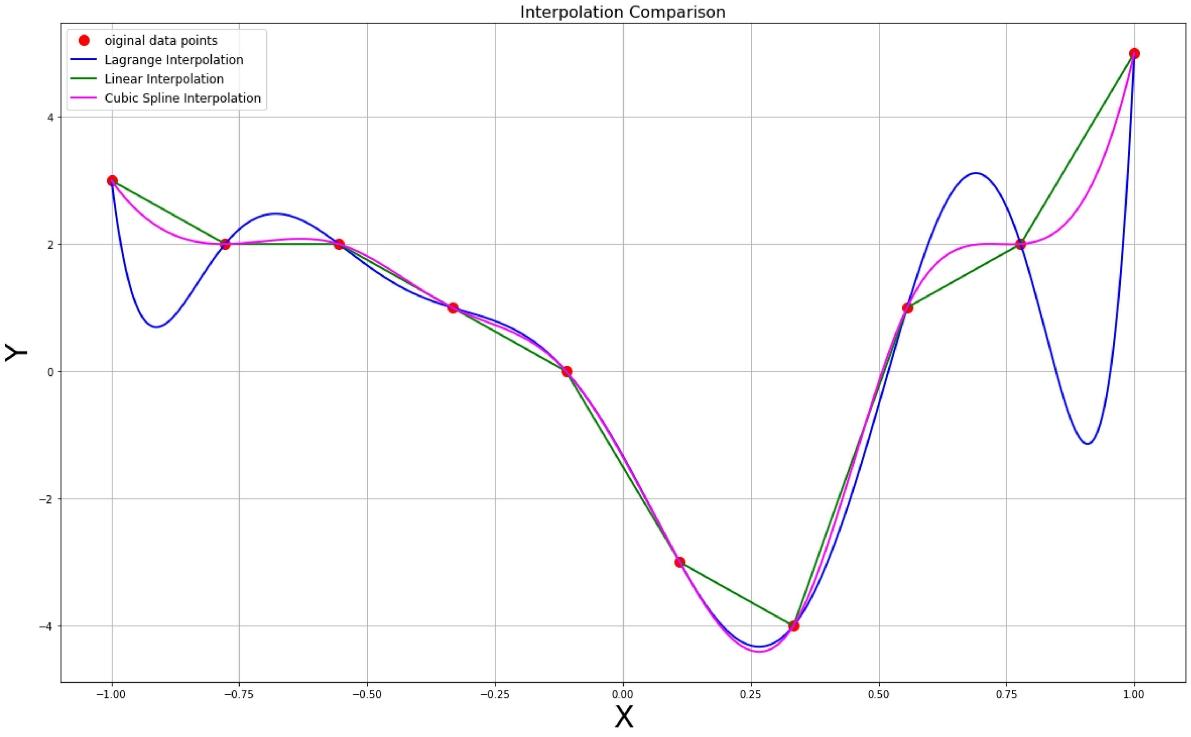
In [8]:
```
# Cubic spline interpolation
cubic_interp = interp1d(xi, yi, kind='cubic')
spline_y = cubic_interp(x)
```

In [9]:
```
# plot & comparison

plt.figure(figsize=(20, 12))

plt.plot(xi, yi, 'ro', markersize=10, label='oiginal data points')
plt.plot(x, lagrange_y, 'blue', linewidth=2, label='Lagrange Interpolation')
plt.plot(x, linear_y, 'green', linewidth=2, label='Linear Interpolation')
plt.plot(x, spline_y, 'magenta', linewidth=2, label='Cubic Spline Interpolatio

plt.legend(fontsize=12)
plt.xlabel('X', fontsize=30)
plt.ylabel('Y', fontsize=30)
plt.title('Interpolation Comparison', fontsize=16)
plt.grid(True)
plt.show()
```


Interpolation Comparison
● oiginal data points
Lagrange Interpolation
Linear Interpolation
Cubic Spline Interpolation
Y
X