

Numerical Techniques Laboratory

Assignment 4.1 | Tanishq Jasoria | 16MA20047

Solve the ODE below by newton-linearization scheme

$$y'' - (y')^2 - y^2 + y + 1 = 0$$

In [1]:

```
import matplotlib.pyplot as plt
import numpy as np
%matplotlib inline
```

In [2]:

```
def get_a(x, h):
    return 1/(h**2) - 1/(4.0 * h**2)
def get_b(x, h):
    return -2/(h**2) + B(x)
def get_c(x, h):
    return 1/(h**2) + A(x)/(2.0 * h)
```

In [3]:

```
epsilon = 0.00001
def ThomasAlgorithm(a, b, c, d, n):
    c_dash = np.zeros(n-1)
    d_dash = np.zeros(n-1)
    c_dash[0] = c[0]/b[0]
    d_dash[0] = d[0]/b[0]
    for itr in range(1, n-1):
        c_dash[itr] = c[itr] / (b[itr] - a[itr] * c_dash[itr-1])
        d_dash[itr] = (d[itr] - a[itr]*d_dash[itr-1]) / (b[itr] - a[itr] * c_dash[i

    y = np.zeros(n-1)
    y[n-2] = d_dash[n-2]

    for itr in reversed(range(n-2)):
        y[itr] = d_dash[itr] - c_dash[itr] * y[itr+1]

    return y
```

In [4]:

```

x0 = 0
xn = np.pi
y0 = 0.5
yn = -0.5
def func(x0, xn, h = 0.1):
    lst = np.arange(x0, xn, h)
    lst = np.append(lst, xn)
    return lst

def BVP(x0, xn, y0, yn, step, epsilon = 0.0001):
    '''Keeping the initialization y = 0.5cos(x)'''
    x = func(x0, xn, step)
    print(x)
    y = 0.5*(np.cos(func(x0, xn, step)))
    print(y.shape)
    delta_y = np.ones(y.shape)
    while(np.amax(np.absolute(delta_y))>epsilon):
        a = [1/step**2 + 2*(y[i+1] - y[i-1])/(4*step**2) for i in range(1, len(y)-1)]
        b = [-2/step**2 - 2*y[i] + 1 for i in range(1, len(y)-1)]
        c = [1/step**2 - 2*(y[i+1] - y[i-1])/(4*step**2) for i in range(1, len(y) - 1)]
        d = [y[i]**2 - y[i] - 1 + (y[i+1] - y[i-1])**2/(4*step**2) - (y[i-1] - 2*y[i]
        delta_y = ThomasAlgorithm(a, b, c, d, len(y)-1)
        delta_y = np.insert(delta_y, 0, 0)
        delta_y = np.append(delta_y, 0)
        y = y + delta_y
    print(y)
    return y

y_new = BVP(x0, xn, y0, yn, step=0.2, epsilon = 0.001)
print(y_new)

```

```

[0.          0.2          0.4          0.6          0.8          1.
 1.2          1.4          1.6          1.8          2.          2.2
 2.4          2.6          2.8          3.          3.14159265]
(17,)
[ 0.5          0.33989476  0.15975631 -0.03130433 -0.22421508 -0.410246
95
-0.58126558 -0.72996405 -0.85008419 -0.9366187  -0.98598128 -0.996132
76
-0.96665433 -0.89876267 -0.79526604 -0.66046432 -0.5          ]
[ 0.5          0.33989476  0.15975631 -0.03130433 -0.22421508 -0.410246
95
-0.58126558 -0.72996405 -0.85008419 -0.9366187  -0.98598128 -0.996132
76
-0.96665433 -0.89876267 -0.79526604 -0.66046432 -0.5          ]

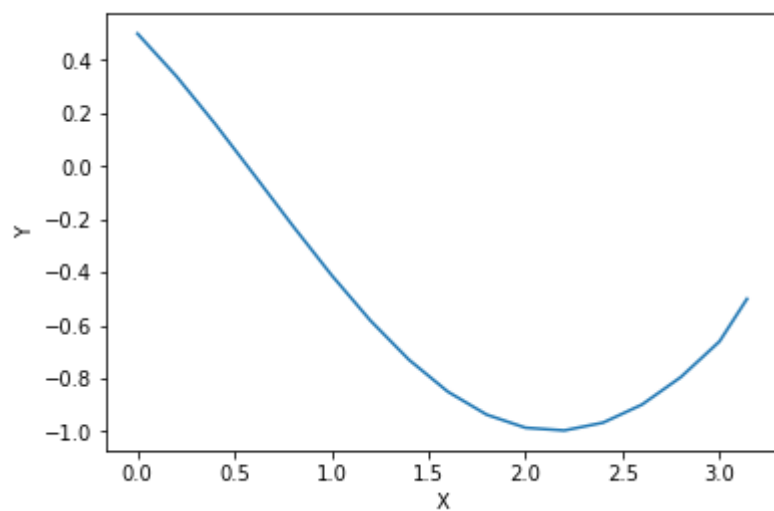
```

In [5]:

```
x = func(x0, xn, 0.2)
plt.xlabel('X')
plt.ylabel('Y')
plt.plot(x, y_new, '-')
```

Out[5]:

[<matplotlib.lines.Line2D at 0x7f6ea34b9390>]



In []: