Solve the ODE for the given conditions

$$y'' = 2 + y^2$$

For the conditions

$$y(0) = 0 = y(1)$$

In [1]:

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

In [2]:

```
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[it

    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 [11]:

```
x0 = 0
xn = 1
y0 = 0
yn = 1
def func(x0, xn, h = 0.1):
    lst = np.arange(x0, xn, h)
    lst = np.append(lst, xn)
    return lst
```

In [29]:

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

    return y
```

In [30]:

```
y_new = BVP(x0, xn, y0, yn, step=0.02, epsilon = 0.0001)
print(y_new)
```

```
[0.     0.0196 0.0384 0.0564 0.0736 0.09   0.1056 0.1204 0.1344 0.1476
 0.16   0.1716 0.1824 0.1924 0.2016 0.21   0.2176 0.2244 0.2304 0.2356
 0.24   0.2436 0.2464 0.2484 0.2496 0.25   0.2496 0.2484 0.2464 0.2436
 0.24   0.2356 0.2304 0.2244 0.2176 0.21   0.2016 0.1924 0.1824 0.1716
 0.16   0.1476 0.1344 0.1204 0.1056 0.09   0.0736 0.0564 0.0384 0.0196
 0.    ]
[ 0.         -0.03824182 -0.07488408 -0.10992806 -0.14337572 -0.175229
66
 -0.20549297 -0.23416918 -0.26126215 -0.28677599 -0.31071497 -0.333083
49
 -0.35388595 -0.37312674 -0.39081016 -0.40694035 -0.42152127 -0.434556
62
 -0.44604985 -0.45600406 -0.46442201 -0.47130609 -0.47665828 -0.480480
15
 -0.48277281 -0.48353696 -0.48277281 -0.48048015 -0.47665828 -0.471306
09
 -0.46442201 -0.45600406 -0.44604985 -0.43455662 -0.42152127 -0.406940
35
 -0.39081016 -0.37312674 -0.35388595 -0.33308349 -0.31071497 -0.286775
99
 -0.26126215 -0.23416918 -0.20549297 -0.17522966 -0.14337572 -0.109928
06
 -0.07488408 -0.03824182  0.        ]
[ 0.         -0.00130437 -0.00260814 -0.00390958 -0.00520603 -0.006493
96
 -0.00776917 -0.00902686 -0.0102618  -0.0114684  -0.01264082 -0.013773
1
 -0.01485923 -0.01589322 -0.01686922 -0.01778158 -0.01862489 -0.019394
1
 -0.02008451 -0.02069187 -0.0212124  -0.02164285 -0.02198051 -0.022223
24
 -0.02236949 -0.02241835 -0.02236949 -0.02222324 -0.02198051 -0.021642
85
 -0.0212124  -0.02069187 -0.02008451 -0.0193941  -0.01862489 -0.017781
58
 -0.01686922 -0.01589322 -0.01485923 -0.0137731  -0.01264082 -0.011468
4
 -0.0102618  -0.00902686 -0.00776917 -0.00649396 -0.00520603 -0.003909
58
 -0.00260814 -0.00130437  0.        ]
[ 0.00000000e+00 -2.55544394e-06 -5.11016655e-06 -7.66200839e-06
 -1.02073842e-05 -1.27413067e-05 -1.52574256e-05 -1.77480863e-05
 -2.02044099e-05 -2.26163952e-05 -2.49730452e-05 -2.72625154e-05
 -2.94722839e-05 -3.15893402e-05 -3.36003898e-05 -3.54920717e-05
 -3.72511831e-05 -3.88649077e-05 -4.03210427e-05 -4.16082187e-05
 -4.27161080e-05 -4.36356170e-05 -4.43590563e-05 -4.48802872e-05
 -4.51948387e-05 -4.52999931e-05 -4.51948387e-05 -4.48802872e-05
 -4.43590563e-05 -4.36356170e-05 -4.27161080e-05 -4.16082187e-05
 -4.03210427e-05 -3.88649077e-05 -3.72511831e-05 -3.54920717e-05
 -3.36003898e-05 -3.15893402e-05 -2.94722839e-05 -2.72625154e-05
 -2.49730452e-05 -2.26163952e-05 -2.02044099e-05 -1.77480863e-05
 -1.52574256e-05 -1.27413067e-05 -1.02073842e-05 -7.66200839e-06
 -5.11016655e-06 -2.55544394e-06  0.00000000e+00]
[ 0.         -0.01994874 -0.03909733 -0.0574453  -0.07499195 -0.091736
36
 -0.1076774  -0.1228138  -0.13714416 -0.15066701 -0.16338077 -0.175283
```

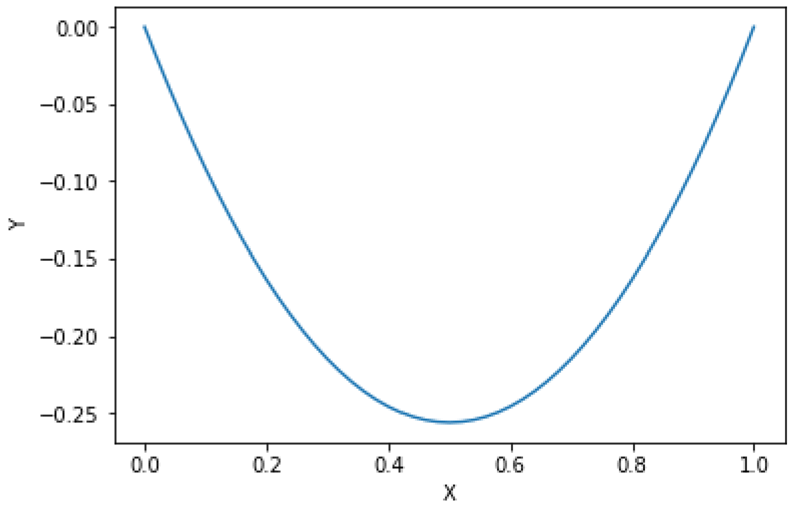
```
85
 -0.18637465 -0.19665155 -0.20611298 -0.21475742 -0.22258341 -0.229589
59
 -0.23577468 -0.24113753 -0.24567713 -0.24939258 -0.25228315 -0.254348
27
 -0.2555875  -0.25600061 -0.2555875  -0.25434827 -0.25228315 -0.249392
58
 -0.24567713 -0.24113753 -0.23577468 -0.22958959 -0.22258341 -0.214757
42
 -0.20611298 -0.19665155 -0.18637465 -0.17528385 -0.16338077 -0.150667
01
 -0.13714416 -0.1228138  -0.1076774  -0.09173636 -0.07499195 -0.057445
3
 -0.03909733 -0.01994874  0.        ]
```

In [31]:

```
x = func(x0, xn, 0.02)
plt.xlabel('X')
plt.ylabel('Y')
plt.plot(x, y_new, '-')
```

Out[31]:

```
[<matplotlib.lines.Line2D at 0x112561eb8>]
```



In [ ]: