notes_2020_11_13

November 25, 2020

1 MATH 210 Introduction to Mathematical Computing

- Numerical methods for second order equations

[1]:
```
import numpy as np
import matplotlib.pyplot as plt
```

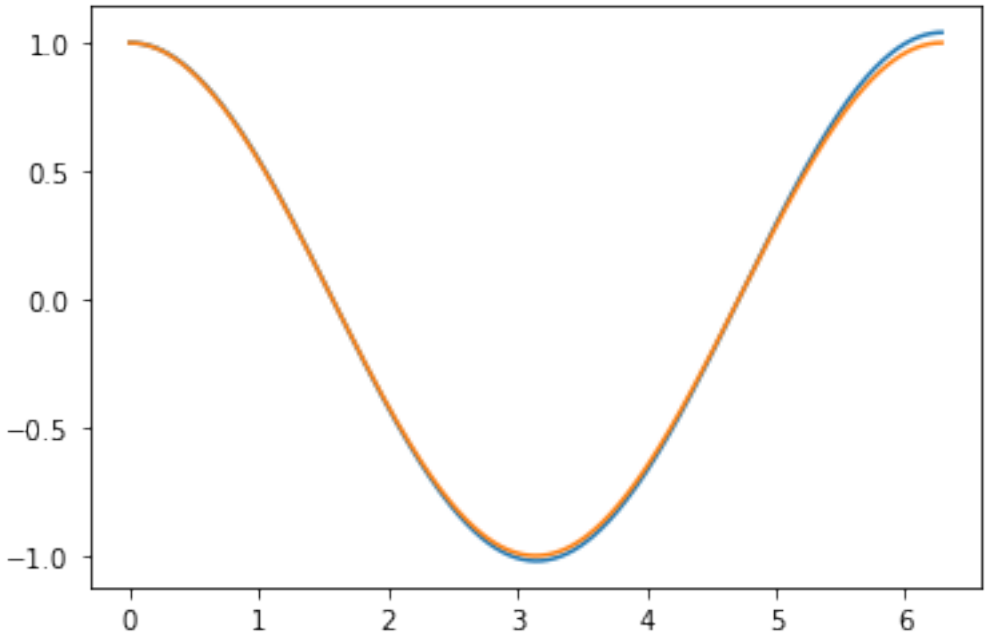
Write a function called `odeEuler2` which takes input parameters `a`, `b`, `c`, `F`, `t`, `y0` and `v0`, and returns the array `y` of y given by Euler's method applied to

$$ay'' + by' + cy = F(t)$$

[2]:
```
def odeEuler2(a,b,c,F,t,y0,v0):
    y = np.zeros(len(t))
    dy = np.zeros(len(t))
    y[0] = y0
    dy[0] = v0
    for n in range(0,len(t)-1):
        h = t[n + 1] - t[n]
        y[n + 1] = y[n] + dy[n]*h
        dy[n + 1] = dy[n] + (F(t[n]) - b*dy[n] - c*y[n])/a*h
    return y
```


Find a simple example where we know the exact solution so that we can verify our function is correct.

[3]:
```
# y'' + y = 0 , y(0) = 1 , y'(0) = 0 => y(t) = cos(t)
a = 1
b = 0
c = 1
F = lambda t: 0
y0 = 1
v0 = 0
t = np.linspace(0,2*np.pi,500)
y = odeEuler2(a,b,c,F,t,y0,v0)
plt.plot(t,y,t,np.cos(t))
plt.show()
```



Success! Now let's try more examples!

[4]:
```
a = 5
b = 2
c = 10
F = lambda t: 0
y0 = 2
v0 = 0
t = np.linspace(0,10,1000)
y = odeEuler2(a,b,c,F,t,y0,v0)
plt.plot(t,y)
plt.show()
```