```
pip install solver
```

```
Requirement already satisfied: solver in /usr/local/lib/python3.7/dist-packages (0.0.4)
```

```
def euler(t,h,y,dy,Func):
  d2y = Func(t,y,dy)
  y_next = y + (h * dy)
  dy_next = dy + (h * d2y)
  return (y_next, dy_next)


def cauchy_euler(params, Func):
  # initial condition
  t0 = params['t0']
  t_akhir = params['t_akhir']
  h = params['h']
  y0 = params['y0']
  dy0 = params['dy0']

  res_euler = []
  t = []
  step = int((t_akhir - t0) / h)

  for i in range(step):
    tm = (i + 1) * h
    (y_next, dy_next) = euler(tm, h, y0, dy0, Func)
    res_euler.append(y_next)
    t.append(tm)
    y0 = y_next
    dy0 = dy_next

  return(t, res_euler)
```

```
from solver import *
```

✓ 0s completed at 10:10 PM ● ✕

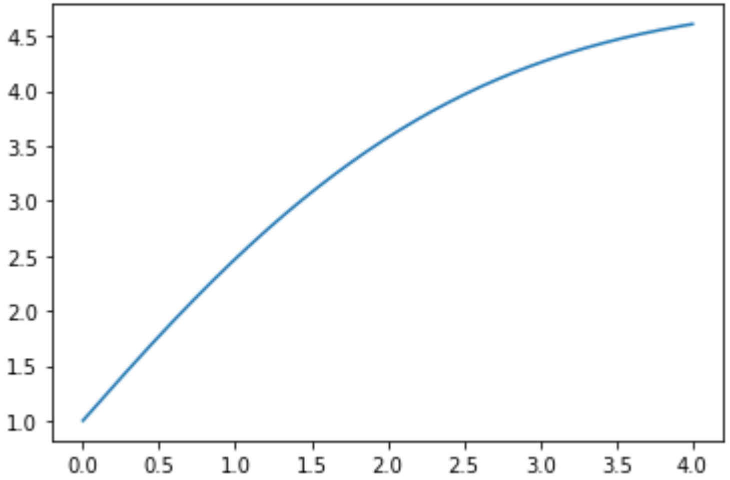
```python
import math

def pendulumfunc(g, l, a):
  return -(g/l) * math.sin(a)

parameter = {
  'g' : 9.8,
  'y0' : 1,
  't0' : 0,
  't_akhir' : 4,
  'h' : 0.001,
  'dy0' : 0.5 * 3.14
}


t, res = cauchy_euler(parameter, pendulumfunc)

plt.plot(t,res)
plt.show()
```



```python
import matplotlib.pyplot as plt
```

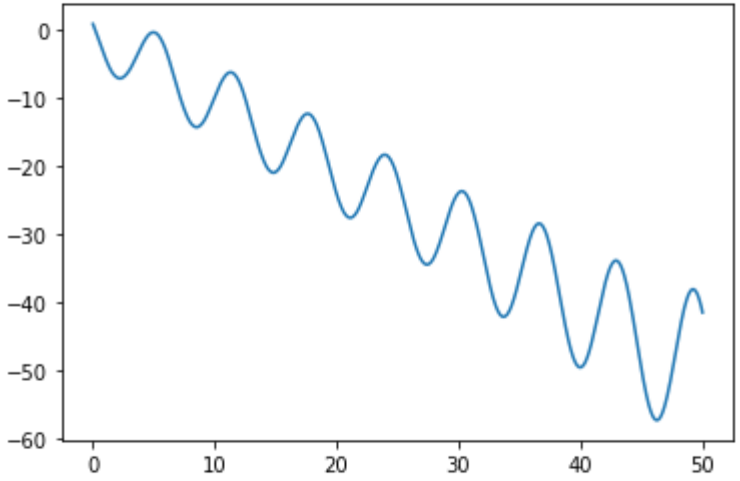
```
from solver import *
import math

def pd2(y, dy, x):
    return -y - dy - (math.sin(x) ** 2)

parameter1 ={
    't0' : 0,
    't_akhir' : 50,
    'h' : 0.05,
    'y0' : 1,
    'dy0' : -9/2
}

x, y = cauchy_euler(parameter1, pd2)

plt.plot(x,y)
plt.show()
```


0
−10
−20
−30
−40
−50
−60
0
10
20
30
40
50