```
import sys
import matplotlib.pyplot as plt
import numpy as np
import os
import PIL
from PIL import Image
#from PIL import ImageOps
## -----------definition de smooth----------------
def smooth(x,window_len=11,window='hanning'):

    if x.ndim != 1:
        raise ValueError, "smooth only accepts 1 dimension arrays."

    if x.size < window_len:
        raise ValueError, "Input vector needs to be bigger than
window size."


    if window_len<3:
        return x


    if not window in ['flat', 'hanning', 'hamming', 'bartlett',
'blackman']:
        raise ValueError, "Window is on of 'flat', 'hanning',
'hamming', 'bartlett', 'blackman'"


    s=np.r_[x[window_len-1:0:-1],x,x[-1:-window_len:-1]]
    #print(len(s))
    if window == 'flat': #moving average
        w=np.ones(window_len,'d')
    else:
        w=eval('np.'+window+'(window_len)')

    y=np.convolve(w/w.sum(),s,mode='valid')
    return y
## ----------- FIN de  definition de smooth----------------
s = []
l = range(0, 255, 25)
l.append(255)

filename = 'phase_level_'

#zoom_p=profile[573, 820]

max_x = np.zeros([12])
j=0

px = (np.pi*6.)/356.

for i in l:
    tab=np.loadtxt('phase_level_'+str(i)+'.txt')
    zoom=tab[470:529]         # zoom sur maximum
```

```
    zoom = smooth(zoom,window='flat')


    [max_x_temp] = np.unravel_index(zoom.argmax(), zoom.shape)
    max_x[j] = max_x_temp    # on trouve le coordonné x de la valeur
max
    j=j+1

#max_x = np.asarray([577, 577, 574, 573, 572, 570, 572, 563, 568,
564, 561, 562])
#yerr = np.asarray([1, 1, 1, 2, 3, 2, 1, 2, 3, 1, 1, 1])
plt.plot(l, -(max_x-max_x[0])*px/np.pi, "o-")

plt.clf()
plt.ylabel(r"$\phi/\pi$ ""decalage de phase  ")
plt.xlabel(" Niveaux de gris ")
plt.xlim(0, 255)
plt.plot(l,-(max_x-max_x[0])*px/np.pi,'o-')

plt.show()
```