

```

import numpy as np
from os import chdir
chdir("D:\\Latex\\Devoirs\\mpsi20152016\\")

def lecture(fichier):
    mon_fichier_lu = open(fichier,'r')
    ListeTemps = []
    ListeTension = []
    for L in mon_fichier_lu:
        if not 'Temps' in L:
            t = float(L.split(';')[0])
            Tension = float(L.split(';')[1])
            ListeTemps.append(t)
            ListeTension.append(Tension)
    mon_fichier_lu.close()
    return(ListeTemps,ListeTension)

def points_d_annulation(tensions):
    listepos=[]
    for i in range(len(tensions)-1):
        if tensions[i] == 0 or tensions[i]*tensions[i+1] < 0:
            if len(listepos)==0 or i>=listepos[-1]+10:
                listepos.append(i)
    return(listepos)

def maxima_successifs(tensions):
    listemax=[]
    listezero = points_d_annulation(tensions)
    for ind in range(len(listezero)//2-1):
        i0 = listezero[2*ind]
        i = i0
        maxi = tensions[i0]
        for k in range(i0, listezero[2*ind+2]):
            if tensions[k]>maxi:
                maxi = tensions[k]
                i = k
        listemax.append(i)
    return(listemax)

def pseudo_periode_par_zeros(temps,tensions):
    listezero = points_d_annulation(tensions)
    listetempszero = [temps[i] for i in listezero]
    listeecart=[listetempszero[i+1]-listetempszero[i]
                for i in range(len(listetempszero)-1)]
    #print(listeecart)
    return(2*(listetempszero[10]-listetempszero[0])/10)

def pseudo_periode_par_maxima(temps,tensions):
    listemax = maxima_successifs(tensions)
    listetempsmax = [temps[i] for i in listemax]
    listeecart=[listetempsmax[i+1]-listetempsmax[i]

```

```

                                IDLE_tmp_ktd52n
                                for i in range(len(listetempsmax)-1)]
                                #print(listeecart)
                                return((listetempsmax[6]-listetempsmax[0])/6)

def decrement_logarithmique(tensions):
    listemax = maxima_successifs(tensions)
    un = tensions[listemax[0]]
    unm = tensions[listemax[3]]
    delta = (1/3)*np.log(un/unm)
    return(delta)

def Q_et_w0(temps,tensions):
    T = pseudo_periode_par_zeros(temps,tensions)
    Omega = 2*np.pi/T
    delta = decrement_logarithmique(tensions)
    X = np.pi/delta #racine de Q^2 - 1/4
    Q = np.sqrt(X**2+1/4)
    w0 = Q*Omega/X
    return(Q,w0)

def Q_et_w0parmax(temps,tensions):
    T = pseudo_periode_par_maxima(temps,tensions)
    Omega = 2*np.pi/T
    delta = decrement_logarithmique(tensions)
    X = np.pi/delta #racine de Q^2 - 1/4
    Q = np.sqrt(X**2+1/4)
    w0 = Q*Omega/X
    return(Q,w0)

```