

```

"""
Série 1 : Rappel
Date: 07-09-2023
Groupe : Ipein/SM4/GB
"""
from math import sqrt

# Exercice 1
## Q1
"""
def init(L):
    L = [1] * 100
    return L

## Prog principal
L = [] #list()
L = init(L)
print(L)

def init(L):

    for i in range(100):
        L.append(1)
    return L

L = []
L=init(L)
print(L)

def init(L):#L : local
    print("id(L) local avant [1]*100 :",id(L))
    L = [1] * 100
    print("id(L) local après [1]*100 :",id(L))
    return L

L = [] # global
print("id(L) gloabl avant init:",id(L))
L = init(L)
print("id(L) gloabl après init:",id(L))
print(L)
"""
def init(L):
    for i in range(100):
        L.append(1)

## Q2
def multiple(L,i):
    ''' met à 0 les elts dont l'indice est % i'''
    for j in range(i+1,len(L)):
        if j % i == 0:
            L[j] = 0

    #return None

## Q3
def suivant(L,i):
    ''' retourne l'indice de premier elt =1 à partir i+1'''
    return L.index(1,i+1)

## Q4
def crible(L):
    init(L)
    i = 2
    while i <= int(sqrt(len(L))):
        multiple(L,i)
        i = suivant(L,i)

    return [i for i in range(2,len(L)) if L[i]==1]

## Q5
def crible_rec(L,i):
    if i <= int(sqrt(len(L))):
        multiple(L,i)
        i = suivant(L,i)
        return crible_rec(L,i)
    else:
        return [i for i in range(2,len(L)) if L[i]==1]

## Q7
L = []
init(L)
p = crible_rec(L,2)

```

```
print(p)
```