

```

# ===== #
# TRI A BULLE (version récursive) #
# Programme de base avec temps de calcul. #
# Version du 25/11/2015 #
# ===== #

# Importations
from random import randint
from time import perf_counter

# La fonction de tri
def tri_bulle_rec(L):
    k = len(L)
    # Liste vide ou ne contenant qu'un élément
    if k <= 1:
        return L
    else:
        Done = True
        for i in range(k-1):
            if L[i] > L[i+1]:
                L[i], L[i+1] = L[i+1], L[i]
                Done = False
        # La liste est triée : pas d'appel récursif
        if Done:
            return L
        # Il y a eu au moins une inversion : appel récursif
        else:
            return tri_bulle_rec(L[:-1]) + [L[-1]]

# Génération d'une liste de nombres entiers aléatoires.
n = int(input('Longueur de la liste à trier ? '))
valmax = int(input('\nValeur maxi des nombres générés ? '))
LNA = [randint(0,valmax) for i in range(n)]

# Tri de la liste LNA des nombres aléatoires + mesure de la durée du tri
t_start = perf_counter()
LT = tri_bulle_rec(LNA)
t_end = perf_counter()
t_ellapse = t_end - t_start

# Affichage des listes (initiale et triée) ... si elles ne sont pas trop grandes ! :)
if n <= 50:
    print('\nListe initiale :')
    print(LNA)
    print('\nListe triée :')
    print(LT)

# Affichage de la durée du tri
print('\nDurée du tri : ' + str(t_ellapse))

# =====
# FIN DU PROGRAMME

```