

Avril - Séance du 02.04. - Corrigé Bogo-Tri

4 séances avant le CCF

Bogo-tri à terminer

Exercice 47 (Bogo tri).

On désire trier un tableau en utilisant la méthode appelée "bogo-tri" (ou tri stupide). Pour effectuer ce tri, on répète, jusqu'à ce que le tableau soit trié, les deux étapes suivantes :

- On vérifie si les éléments sont classés par ordre croissant.
 - Si ils ne le sont pas, on mélange aléatoirement les éléments du tableau.
- 1) Écrire en pseudo-code et en Python une fonction `ordre` qui prend en argument un tableau et retourne Vrai si les éléments du tableau sont triés dans l'ordre croissant, et Faux sinon.
 - 2) On considère la fonction `F` suivante :

```
1 Fonction F(tab)
2 Variables locales : ...
3 DEBUT FONCTION
4     bogotab ← []
5     Tant Que longueur(tab)>0:
6         i ← randint(0, longueur(tab)-1)
7         bogotab ← bogotab + [tab[i]]
8         tab ← tab[0:(i-1)] + tab[(i+1):(longueur(tab)-1)]
9     Fin Tant Que
10    Retourner : bogotab
11 FIN FONCTION
```

- a) Lister les variables utilisées dans la fonction `F`, ainsi que leur type.
 - b) À quoi sert la ligne 8 de la fonction ? Que se passe-t-il sans cette ligne ?
 - c) Que fait la fonction `F` ?
 - d) Traduire cette fonction en Python.
- 3) Écrire en pseudo-code et en Python une fonction `tri` qui prend en paramètre un tableau et retourne le tableau trié grâce au bogo-tri.
Cette fonction pourra utiliser les fonctions `F` et `ordre`.
Attention à ne tester que sur de petits tableaux.
 - 4) Pour que le temps de tri soit raisonnable, on décide de limiter le nombre d'itérations du bogo-tri à 100 000. Si au bout de 100 000 itérations, le tableau n'est toujours pas trié, on affichera un message d'échec.
Écrire en Python une fonction `tri2` reprenant la fonction `tri` mais avec la limite d'itérations.

Solution d'Edouard

```

from random import *

def ordre(tab):
    Verif = True
    for i in range(len(tab)-1):
        if tab[i+1] < tab[i] :
            Verif = False
    return(Verif)

def F(tab):
    bogotab = []
    while len(tab) > 0:
        i = randint(0, len(tab)-1)
        bogotab += [tab[i]]
        tab = tab[0:i] + tab[(i+1) : len(tab)]
    return (bogotab)

def Tri(tableau):
    while ordre(tableau) != True:
        tableau = F(tableau)
    return(tableau)

def Tri2(tableau):
    compteur = 0
    while ordre(tableau) != True and compteur < 100001:
        tableau = F(tableau)
        compteur = compteur + 1
    return(tableau)

table = []
n = int(input("Taille du tableau : "))
for i in range(n):
    table.append(int(input()))
print(table)
tableauF = Tri2(table)
if ordre(tableauF):
    print(tableauF)
else :
    print("Echec du tri après 100000 essais")

```