

Dernière mise à jour	Informatique	Denis DEFAUCHY – Site web
31/05/2023	8 - Tris	TD 8-5 – Tri par comptage

Informatique

8 Tris

TD 8-5
Tri par comptage

Dernière mise à jour	Informatique	Denis DEFAUCHY – Site web
31/05/2023	8 - Tris	TD 8-5 – Tri par comptage

Exercice 1: Tri par comptage

Soit une liste L de n entiers dans l'intervalle $[0, max]$, max n'étant pas connu à l'avance. Il est possible de trier L en la parcourant une première fois pour trouver son maximum, puis une seule fois supplémentaire pour dénombrer ses éléments et reconstruire une nouvelle liste triée de ses entiers. On veillera à créer des codes de complexité linéaire.

Question 1: Créer une fonction `Max(L)` renvoyant le maximum de L

Question 2: Créer une fonction `Denombre(L,m)` renvoyant une liste `Res` de $m+1$ termes où `Res[i]` est le nombre de fois qu'apparaît i dans L , m étant le maximum de L

Exemple :

```
>>> Denombre([1,1,1,3,3,2,2,2,2,5],5)
[0, 3, 4, 2, 0, 1]
```

Question 3: Créer une fonction `Construction(L)` créant une liste contenant dans l'ordre, $L[0]$ fois l'entier 0, $L[1]$ fois l'entier 1, [...] $L[i]$ fois l'entier i ...

Exemple :

```
>>> Construction([0, 3, 4, 2, 0, 1])
[1, 1, 1, 2, 2, 2, 2, 3, 3, 5]
```

Question 4: Créer la fonction `f_tri_comptage(L)` réalisant ce tri – Veillez à traiter le cas d'une liste vide en renvoyant une liste vide

Remarques :

- Vérifiez que vous dénombrez bien les valeurs de 0 à max (soit $max+1$ entiers) !
- Ce tri est très efficace, car nous savons quoi rechercher (des entiers dans un intervalle donné...). Sa complexité est en linéaire. Ce n'est pas la même chose quand on ne sait pas à priori quelles valeurs on recherche...

Le tri que nous venons de mettre en place recrée des éléments (entiers) qui n'ont rien à voir « physiquement » avec les éléments de la liste L initiale. On ne peut donc pas parler de stabilité. Il est toutefois possible d'envisager de créer un tri par dénombrement stable ainsi :

- Trouver comme précédemment le maximum m de la liste d'entiers positifs L
- Créer une liste `Res` de $m+1$ listes vides
- Stocker dans `Res[e]` les éléments de L valant e dans leur ordre d'apparition dans L
- Recréer la liste L triée des vrais éléments de L

Question 5: Créer la fonction `f_tri_comptage_stable(L)` de complexité linéaire

Remarque : pour vérifier que ce tri est stable, nous ajouterons des flottants du type 1.0 dans L , pensez donc à transformer vos indices en entiers là où c'est nécessaire afin que cela fonctionne

Vérifier :

```
>>> f_tri_comptage_stable([3,3.0,1.0,1,5])
[1.0, 1, 3, 3.0, 5]
```