TP7 : deques/tri par paquets/ Fibonacci

EXO1

Afin de manipuler correctement les deques il est demandé de programmer les différentes fonctions vues en TD dans l'exos sur les TODO listes.

1) Base de travail

Ecrire les fonctions de base de cette structure de données.



- Ajout_en_tete
- Ajout_en_fin
- Retrait_en_tete
- Retrait_en_fin

2) Fonctions de ma TODO liste

Nous avions énoncé quelques contraintes:



- Je veux pouvoir mettre une action à réaliser en priorité absolue
- Je veux pouvoir, quand je prends une action à réaliser, la comparer avec la suivante et choisir entre les deux et remettre celle qui je ne fais pas en tête de liste, ou si vraiment je ne me sens pas une envie débordante de la réaliser en fin de liste
- Vérifier la dernière tâche ajoutée

Il faut donc se servir des fonctions définies au 1) pour écrire

```
def ajout(toDoListe,action, priorite):
'''cette fonction ajoute l'action dans la toDoListe au bon endroit selon la priorite (0 ou 1)'''
```

```
def done(toDoListe,choix):
''' cette fonction permet de retirer une action de la toDoListe, soit systématiquement la première de la liste (choix=0), soit de choisir entre entre les deux premières (choix=1). Il faut alors afficher les actions et demander à l'utilisateur de choisir. Une fois son choix fait il doit dire si il veut mettre l'action qu'il ne choisit pas à sa place (au début de la liste) ou si vraiment il ne veut la faire rapidement et veut la mettre à la
```

```
fin. Il faut bien sûr effectuer cette opération.
```

```
def  verif(toDoListe):
''' cette fonction permet d'aller voir la dernière action stockée et de la remettre après'''
```

faire un petit programme qui permet de vérifier toutes ces fonctions.

EXO2

Tri par Paquets (vu en TD7) Principe:soit une liste de nombres réels on va créer une liste de n listes (autant que de données dans la liste initiale , découpant l'intervalle des données en n intervalles) et répartir ces données dans les petites listes(paquets). On prendra ensuite un algorithme de tri à bulle pour trier chaque paquet puis les concaténer.



- Le tri à bulles trie sur place
- Le tri par paquets ne trie pas sur place
- **Qu'est-ce que cela engendre au niveau de la programmation?

**

Il faut donc deux fonctions

```
def tri_bulle(T):
```

```
def tri_par_paquet(T):
```

On veut comparer les temps d'exécution des deux tris. Pour cela, ajoutez

```
import time
```

et utiliser

```
 t=time.time()
```

Pour générer la liste utiliser np.random.randn() et ajouter

```
import numpy as np
```

EXO3

Programmer la fonction Fibonnaci de manière récursive et de manière itérative

From:
https://wiki.centrale-med.fr/informatique/ - **WiKi informatique**

Permanent link:
https://wiki.centrale-med.fr/informatique/tc_info:tp7b

Last update: **2019/11/28 09:38**