

T.D. 4

Fonctions de base

On utilise les fonctions suivantes pour les piles et les files :

```
def empiler (e, p):  
    p.append (e)  
  
def depiler (p):  
    p.pop ()  
  
def sommet (p):  
    return p [len (p) - 1]  
  
def enfiler (e, f):  
    f.append (e)  
  
def defiler (f):  
    for i in range (len (f) - 1):  
        f [i] = f [i + 1]  
    f.pop ()  
  
def tete (f):  
    return f [0]
```

Exercice 1: Comparaison de deux piles

Ecrire une fonction qui teste si deux piles contiennent les mêmes éléments du sommet vers la base.

Exercice 2: Inversion d'une pile

Ecrivez une fonction inverser (p) qui inverse l'ordre des éléments d'une pile en utilisant une file.

Exercice 3: Affichage d'une liste en ordre inversé

Ecrire une fonction non récursive qui affiche les éléments d'une liste d'entiers dans l'ordre inverse de leur apparition dans la liste sans modifier la liste initiale mais en utilisant une pile.

Exercice 4: Inversion d'une file

Ecrire une fonction qui inverse une file d'entiers. On utilisera une pile pour faire l'inversion.

Exercice 5: Parenthésage correct

Une liste chaînée de caractères contient des parenthèses. Une liste est correcte si toute parenthèse ouvrante se trouve avant la parenthèse fermante correspondante et si la sous liste entre ces deux parenthèses est aussi correcte. Par exemple `((()()))` est correcte mais `)()` ne l'est pas. Ecrire une fonction qui détermine si une liste est correcte.

Exercice 6: Palindrome

Un palindrome est une chaîne de caractères qui se lit de la même manière de gauche à droite et de droite à gauche. En utilisant une pile et une file, écrire une fonction qui détermine si une liste chaînée de caractères est un palindrome.