

Tri crêpe

Principe

On dispose d'un empilement de crêpes que l'on souhaite les ranger dans l'ordre croissant, la crêpe la plus petite en haut.

Pour celà, vous ne disposez que d'un seul type opération : glisser une spatule sous une crêpe et retourner tout l'empilement de crêpes se trouvant sur votre spatule.

Consigne

Nous utiliserons une liste pour représenter notre empilement de crêpes. Le sommet de l'empilement sera le premier élément de la liste. Les crêpes sont représentées par des nombres, plus son nombre est grand plus la crêpe est grande.

Liste fonctions

- `retourner_crepe(empilement : list) , n : (int)` : retourne n crêpe.
- `plus_grande_crepe(empilement : list, n : int) -> int` : Cherche la plus grande crepe dans l'empilement.
- `placer_crepe_au_sommet(empilement : list, n : int) -> int` : place la crêpe au sommet.
- `tri_crepe(empilement : list)` : réalise le tri crêpe

Implémentation

Commençons par la fonction `retourner_crepe(empilement : list) , n : (int)` qui prend un empilement de crepe, un entier n représentant l'emplacement de la spatule et retourne ainsi les crêpes sur la spatule

```
def retourner_crepe(empilement,n):  
    """  
    Retourne les n crêpes du haut  
    :param empilement: (list) liste représentant l'empilement de crêpes  
    :param n: (int) nombre de crêpes à retourner  
    :return: None  
    : Examples :  
    >>> l = [5, 3, 4, 7, 9, 8, 1, 2]  
    >>> retourner_crepe(l,4)  
    >>> l  
    >>> [7, 4, 3, 5, 9, 8, 1, 2]  
    """  
    pass
```

Désormais, nous allons rechercher l'indice de la plus grande crêpe parmi un certain nombres. La fonction `plus_grande_crepe(empilement : list, n : int) -> int` qui prend un empilement de crepe, un entier n représentant la tranche parmi laquelle on recherche la plus grande crêpe. Cette fonction renverra l'indice dans la liste de la plus grande crêpe.

```
def plus_grande_crepe(empilement,n):
    """
    Renvoie l'indice de la plus grande crêpe parmi les n premières
    :param empilement: (list) liste représentant l'empilement de crêpes
    :param n: (int) nombre de crêpes
    :return: (int) indice de la plus grande crêpe
    : Examples :
    >>> l = [5, 3, 4, 7, 9, 8, 1, 2]
    >>> plus_grande_crepe(l,4)
    >>> 3
    """
    pass
```

Ensuite, nous allons placer une crepe au sommet. La fonction `placer_crepe_au_sommet(empilement : list, n : int) -> int` qui prend un empilement de crepe, un entier n représentant le nombre de crêpe à retourner.

Une méthode serait de chercher la plus grande crêpes parmi les n premières, de retourner toutes les jusqu'à la plus grande, puis de retourner toutes les crêpes.

```
def placer_crepe_au_sommet(empilement,n):
    """
    Place la plus grande crêpe au sommet de la pile de crêpe
    Repère la plus grande crêpe parmi les n premières
    Retourne l'empilement de crêpes en placant la spatule sous cette plus grande crêpe
    Puis retourne les n crêpes du sommet
    :param empilement: (list) liste représentant l'empilement de crêpes
    :param n: (int) nombre de crêpes à retourner
    :return: None
    : Examples :
    >>> l = [5, 3, 4, 7, 9, 8, 1, 2]
    >>> placer_crepe_au_sommet(l,8)
    >>> l
    [2, 1, 8, 5, 3, 4, 7, 9]
    """
    pass
```

Il est temps d'implémenter l'algorithme `tri_crepe(empilement : list)` .

```
def tri_crepe(empilement):
    """
    Effectue le tri crêpe
    :param empilement: (list) liste représentant l'empilement de crêpes
    :return: None
    """
    pass
```

Félicitations, vous avez implémenter le tri crêpe !