

Exercice 6

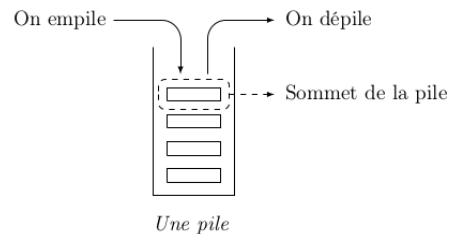
Piles

On dispose d'une classe Pile:

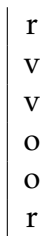
Les primitives de la classe Pile:

- `p = Pile()` → crée la Pile vide `p`.
- `p.est_vide()` → renvoie **True** si la pile est vide.
- `p.empiler(val)` → empile `val` dans `p`.
- `p.depiler()` → dépile `f` et renvoie la valeur.
remarque: Si la pile est vide la méthode `depiler` renvoie **None**

Le principe de la pile



On considère la pile p_1 ci-dessous:



On peut la représenter par cette suite:

(sommet de la pile) `[r]->[v]->[v]->[o]->[o]->[r]`

? QUESTION 1:

p_1 est la pile ci-dessus.

1. Que contiennent les deux piles après exécution de ce programme?

```
p2=Pile()
while not(p1.est_vide()):
    p2.empiler(p1.depiler())
```

p_1 est vide et p_2 contient les valeurs de p_1 dans le sens inverse

2. On considère la pile p : `[v]->[v]->[v]->[o]->[o]->[r]->[r]->[r]->[v]`
Que contient la pile p_2 après exécution de ce programme?

```
p2=Pile()
while not(p.est_vide()):
    c1 = p.depiler()
    c2 = p.depiler()
    c3 = p.depiler()
    if c1 == c2 and c2 == c3:
        pass
    else:
        p2.empiler(c1)
        p.empiler(c3)
        p.empiler(c2)
```

`[v]->[o]->[o]`

Une maîtresse d'école décide d'instaurer un système de points correspondant à la participation des élèves.

À chaque élève, est associée une pile de type LIFO qui mémorise les différentes réponses qu'il a faites.

Si un élève répond correctement à une question, alors un feu vert est empilé sur sa pile.

Si il répond de façon approximative, un feu orange est empilé sur sa pile.

Si il fournit une mauvaise réponse, un feu rouge est empilé sur sa pile.

Un élève a initialement 0 points, puis les points sont comptés de la façon suivante:

- Si la pile contient 3 fois de suite la couleur verte, son nombre de points est incrémenté de 10 points.
- Si la pile contient 3 fois de suite la couleur orange, son nombre de points est incrémenté de 5 points.
- Si la pile contient 3 fois de suite la couleur rouge, son nombre de points est décrémenté de 10 points.

De plus Si trois feux identiques apparaissent au sommet de la pile au cours du comptage, alors les trois feux sont dépilés de la pile.

Voici la pile d'un élève: (R pour rouge, V pour vert et O pour orange)

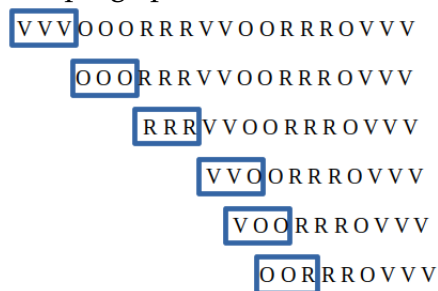
(sommet de la pile) → VVVOOORRRVVOORROVVV

? QUESTION 2:

1. Déterminer le nombre de point obtenu par cet élève.**5**
2. Quel est l'état de la pile après comptage des points?**[v]→[v]→[o]→[o]→[o]**

? QUESTION 3:

On peut schématiser une méthode de parcours de la pile lors du comptage par :



Une autre pile pour garder les sommets non éliminés

(sommet de la pile) → V V

Combien d'étapes reste-t-il avant que la pile soit vide?**4**

? QUESTION 4:

Écrire une fonction en Python qui prend en paramètre la pile des feux d'un élève et qui renvoie le nombre de points obtenus.

On pourra utiliser une autre pile et s'assurer que la pile des feux de l'élève contienne les feux non éliminés lors du comptage des points.

```
def points(p):
    p2=Pile()
    points=0
    while not(p.est_vide()):
        c1=p.depiler()
        c2=p.depiler()
        c3=p.depiler()
        if c1 == c2 and c2 == c3:
            if c1=="r":
                points+=-10
            elif c2=='o':
                points+=5
            else:
                points+=10
        else:
            p.empiler(c3)
            p.empiler(c2)
            p2.empiler(c1)
    for i in range(len(p2)):
        p.empiler(p2.depiler())
    return points
```