

Enoncé

Cet exercice utilise des piles qui seront représentées en Python par des listes (type `list`).

On rappelle que l'expression `liste_1 = list(liste)` fait une copie de `liste` indépendante de `liste`, que l'expression `x = liste.pop()` enlève le sommet de la pile `liste` et le place dans la variable `x` et, enfin, que l'expression `liste.append(v)` place la valeur `v` au sommet de la pile `liste`.

Compléter le code Python de la fonction `positif` ci-dessous qui prend une pile `liste` de nombres entiers en paramètre et qui renvoie la pile des entiers positifs dans le même ordre, sans modifier la variable `liste`.

Script Python

```

1 def positif(pile):
2     pile_1 = ... (pile)
3     pile_2 = ...
4     while pile_1 != []:
5         x = ...
6         if ... >= 0:
7             pile_2.append(...)
8     while pile_2 != ...:
9         x = pile_2.pop()
10        ...
11    return pile_1

```

Exemple :

Script Python

```

>>> positif([-1, 0, 5, -3, 4, -6, 10, 9, -8])
[0, 5, 4, 10, 9]
>>> positif([-2])
[]

```

