

# Correction

NSI - 2021 Sujet Zéro (21-sujet-zero)

## Exercice 1 - Pile de crêpes

1. Le contenu de la pile Q après l'exécution des instructions est :

8
5
2
4

2.1.

```
def hauteur_pile(P):
    Q = creer_pile_vide()
    n = 0
    while not(est_vide(P)): # Compter
        n += 1
        x = depiler (P)
        empiler(Q, x)
    while not(est_vide(Q)): # Rétablir la pile
        x = depiler(Q)
        empiler(P, x)
    return n
```

2.2.

```
def max_pile(P, i):
    """ Recherche de la position du maximum parmi les i derniers éléments
        de la pile P. Le sommet de la pile est 1. P conserve son état d'origine. """
    if est_vide(P):
        return "???"
    j, j_max = 1, 1
    Q = creer_pile_vide()
    val_max = depiler(P)
    i -= 1
    empiler(Q, val_max)
    while not est_vide(P) and i > 0 :
        j += 1
        i -= 1
        val = depiler(P)
        empiler(Q, val)
        if val > val_max:
            val_max = val
            j_max = j

    while not est_vide(Q): # Rétablir la pile P
        empiler(P, depiler (Q))

    return j_max
```

# Correction

NSI - 2021 Sujet Zéro (21-sujet-zero)

3.

```
def retourner(P, j):  
    Q = creer_pile_vide()  
    R = creer_pile_vide()  
    while j > 0: # Retirer les j premiers éléments  
        empiler(Q, depiler(P))  
        j -= 1  
    while not est_vide(Q): # Déplacer dans la pile R  
        empiler(R, depiler(Q))  
    while not est_vide(R): # Remettre dans la pile P  
        empiler(P, depiler(R))
```

4.

```
def tri_crepes(P):  
    for i in range(hauteur_pile(P), 1, -1):  
        j = max_pile(P, i) # Recherche le plus grand  
        retourner(P, j)   # Retourne jusqu'au plus grand  
        retourner(P, i)   # Retourne la pile entière restant à trier
```