

# Structures de données et algorithmes

## Répétition 3: Pile, file et liste

Jean-Michel BEGON - Romain MORMONT

15 mars 2019

### Exercice 1

1. Soient deux nœuds  $n$  et  $m$  de listes simplement liées. Que provoquent les opérations suivantes sur la liste auquel  $n$  appartient (mais pas  $m$ ) ?
  - (a)  $n.next = n.next.next$
  - (b)  $m.next = n.next; n.next = m$
  - (c)  $n.next = m; m.next = n.next$
2. Soit  $L$  une liste simplement liée.
  - (a) Ecrire une fonction `AntePenultimate(L)` permettant de récupérer l'avant-avant dernier nœud de  $L$ .
  - (b) Ecrire une fonction `Remove(L, p)` supprimant le nœud  $p$  de la liste  $L$ .
  - (c) Ecrire une fonction `Reverse(L)` permettant d'inverser (en place) la liste  $L$ .

### Exercice 2

1. Soit une pile  $S$ . Illustrer l'état de  $S$  au fil des instructions.

```
push(S, 5)
push(S, 3)
pop(S)
push(S, 2)
push(S, 8)
pop(S)
pop(S)
push(S, 9)
push(S, 1)
pop(S)
push(S, 7)
push(S, 6)
pop(S)
pop(S)
push(S, 4)
pop(S)
pop(S)
```

2. Soit une file  $Q$ . Illustrer l'état de  $Q$  au fil des instructions.

```
enqueue(Q, 5)
enqueue(Q, 3)
dequeue(Q)
enqueue(Q, 2)
enqueue(Q, 8)
dequeue(Q)
dequeue(Q)
enqueue(Q, 9)
enqueue(Q, 1)
dequeue(Q)
enqueue(Q, 7)
enqueue(Q, 6)
dequeue(Q)
dequeue(Q)
enqueue(Q, 4)
dequeue(Q)
dequeue(Q)
```

### Exercice 3

Concevez une structure de données contenant 2 piles et implémentée avec un seul tableau. Implémentez les opérations `push` et `pop`.

### Exercice 4

Implémenter une structure de pile d'entiers qui dispose également d'une fonction `MS-Max(S)` permettant de consulter la valeur maximum actuellement stockée dans la pile. `MS-Pop` et `MS-Push` doivent être efficaces (i.e.  $\mathcal{O}(1)$ ).

### Exercice 5

Implémenter une file à l'aide de deux piles. Quelle est la complexité des opérations `Enqueue` et `Dequeue` ?