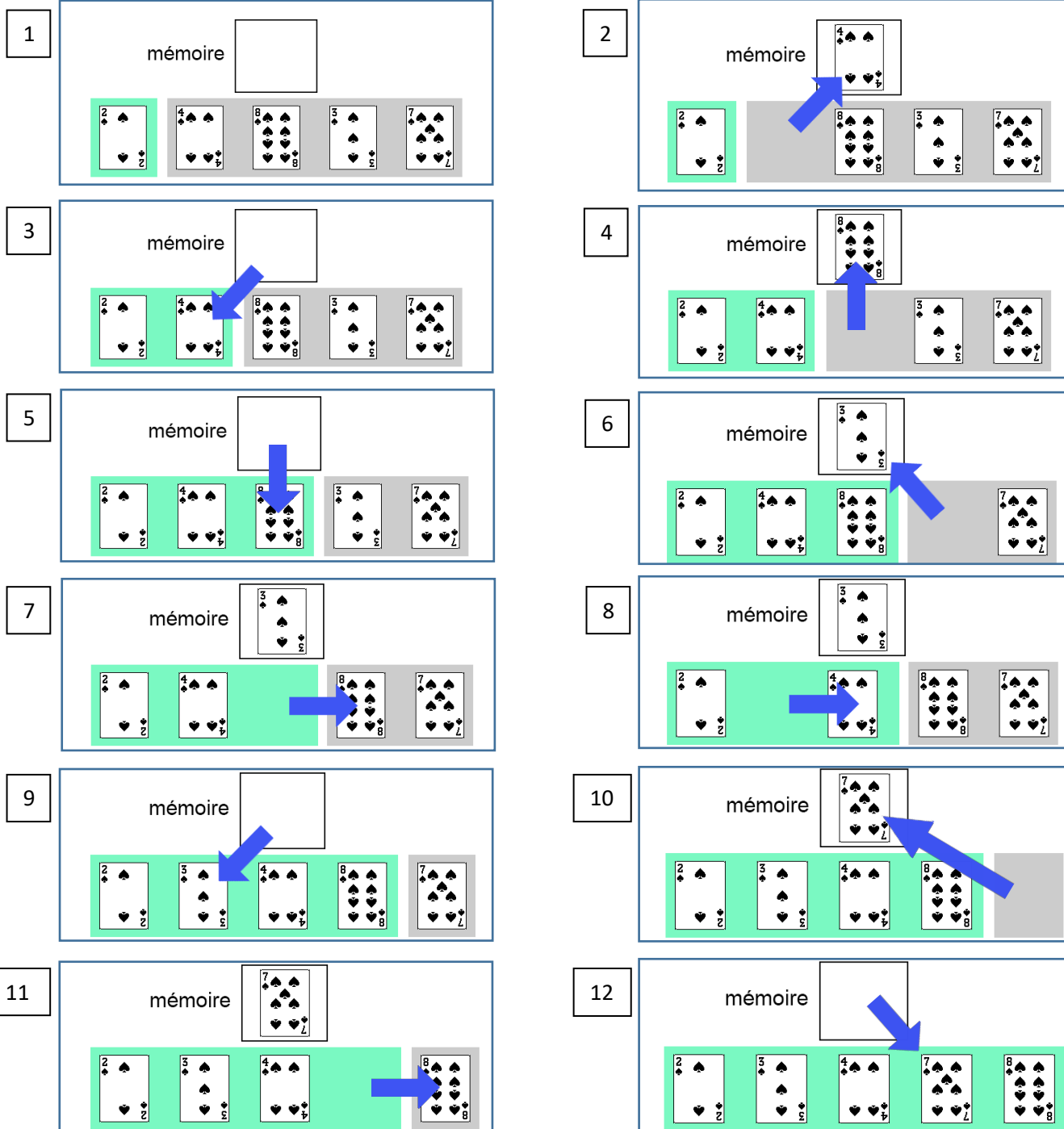
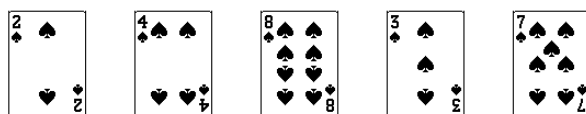


# Exercice sur le tri par insertion

Cet exercice a pour but de vous permettre de retrouver l'**algorithme du tri par insertion**.

Voici un rappel du **principe** de ce tri. On cherche à trier la **suite de cartes** suivantes :



### Exercice 1 :

#### Complétez le principe du tri par insertion d'un tableau d'éléments.

On considère que le **tableau** est composé d'une **partie triée** et d'une **partie non-triée**. La **partie triée** est initialement composée du **premier** élément du tableau.

- Pour trier un **tableau** en utilisant le **tri par insertion**, on commence par traiter le ..... élément du tableau (on considère le ..... élément comme **déjà trié**).
- Cet élément est considéré comme notre "clé" et est placé en .....
- Maintenant, il faut comparer cette clé avec chaque élément ..... dans le tableau, en commençant par l'élément immédiatement ..... et en continuant jusqu'à ce que l'on trouve un élément qui n'est pas plus ..... que la clé, ou jusqu'à ce que l'on atteigne le ..... du tableau.
- Pendant ce processus, chaque élément qui est plus ..... que la clé doit être déplacé vers la .....
- Enfin, une fois que l'on a trouvé la position correcte pour la clé, on la récupère depuis la ..... et on l'insère à sa bonne position.
- On répète cette procédure pour chaque élément du tableau, en allant de ..... à ..... . À la fin, le tableau sera trié par ordre .....

### Exercice 2 :

#### Répondre aux questions suivantes.

On notera **n** la longueur du **tableau** traité.

- a. L'algorithme du tri par insertion parcourt tous les éléments du tableau un par un, on utilisera pour cela une **boucle POUR**. À partir de quel **indice** commencera cette boucle, et à quel **indice** s'arrêtera-t-elle ?  
Complétez l'instruction :

POUR **i** ALLANT DE ... à ... :  
(corps de la boucle POUR)

- b. La « clé » est l'élément actuellement considéré pour l'insertion dans la **partie déjà triée** du tableau. Il s'agit de l'élément d'indice **i** du tableau. Quelle instruction utilisera-t-on pour stocker cet élément en mémoire ?

.....

- c. On aura ensuite besoin d'une autre boucle pour comparer la « clé » avec chaque élément dans la **partie déjà triée** du tableau. Cette boucle doit aller **à l'envers**, en commençant par l'élément **immédiatement avant** la « clé » (d'indice **i**) et en continuant jusqu'à ce qu'elle trouve un élément qui n'est **pas plus grand** que la « clé », ou jusqu'à ce qu'elle atteigne le **début du tableau**. Pendant cette boucle, chaque élément qui est **plus grand** que la « clé » sera **déplacé vers la droite**. On utilisera pour cela une **boucle TANT QUE**. Complétez :

**j** = .....  
TANT QUE ..... ET ..... :  
    tableau[.....] = tableau[.....] (déplacement de l'élément d'indice **j** à sa droite)  
    **j** = **j** - 1 (on passe à l'élément à gauche de **j**)

- d. Enfin, une fois que la **boucle interne** est terminée, on a trouvé la bonne place pour la « clé ». Elle doit ainsi être insérée à la position **juste après** l'élément qui n'était **pas plus grand que la « clé »** (ou au début du tableau si aucun tel élément n'a été trouvé). Quelle instruction permet de replacer la « clé » à la bonne position ?

.....

### Exercice 3 : En vous aidant des deux exercices précédents, écrire l'algorithme du tri par insertion.