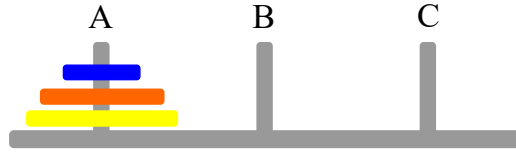


Les tours de Hanoï

Objectif

On souhaite déplacer la tour pyramidale de la position A vers la position C **en un minimum d'étapes**.



Pour résoudre ce problème, on doit respecter les règles ci-dessous :

- ★ on ne peut déplacer qu'une pièce à la fois ;
- ★ on peut seulement placer une pièce sur une pièce plus grande ou sur un emplacement vide.

Manipulations

A l'aide des pièces distribuées, résoudre le problème posé pour une tour de 2 étages, 3 étages, puis 4 étages.

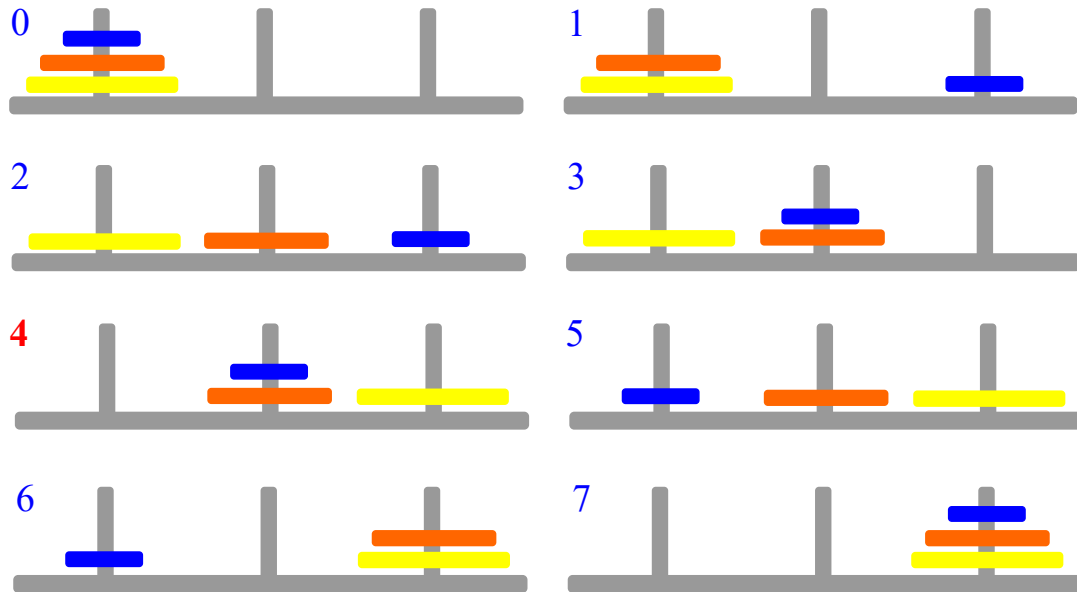
Précisez, pour chaque cas, le nombre d'étapes nécessaires :

- ★ nombre d'étapes pour deux étages :
- ★ nombre d'étapes pour trois étages :
- ★ nombre d'étapes pour quatre étages :

A	B	C

PARTIE B Implémentation

A l'étape 4, on retrouve la tour à deux étages en position C, or nous savons déplacer une tour à 2 étages en 3 étapes :



Essayons maintenant de reproduire dans le cas général ce que l'on vient de faire dans ce cas particulier.

On arrive vite à la conclusion que la résolution du problème pour toute valeur de n est récursive :

- ★ On déplace $n - 1$ disques de la tour initiale à la tour intermédiaire en passant par la tour finale.
- ★ On déplace le plus grand disque de la tour initiale vers la tour finale.
- ★ On déplace $n - 1$ disques de la tour intermédiaire à la tour finale en passant par la tour initiale.

Le cas de base étant le déplacement d'un seul disque.

En Python cela donne cette implémentation :

```
def hanoi(depart, destination, intermediaire, n):  
    if n==1:  
        print("Déplacer une pièce de", depart, " vers ", destination)  
    else:  
        hanoi(depart, intermediaire, destination, n-1)  
        print("Déplacer une pièce de", depart, " vers ", destination)  
        hanoi(intermediaire, destination, depart, n-1)
```

1. Saisir ce programme.
2. Exécuter : `hanoi("A", "C", "B", 3)`, `hanoi("A", "C", "B", 4)`, `hanoi("A", "C", "B", 5)`...
3. Compléter le programme pour qu'il affiche le nombre d'étapes nécessaires. (Utiliser une variable globale).