

Fonctionnement des Piles d'Exécution

Anis Saied

October 27, 2024

1 Pile d'exécution

- Une pile d'exécution est une structure de données qui stocke les appels de fonctions en cours d'exécution.
- Chaque appel de fonction crée un nouveau cadre (frame) dans la pile, contenant les **variables locales** et l'**adresse de retour**.
- Les cadres sont empilés lors des appels et dépilés lorsque les fonctions se terminent.

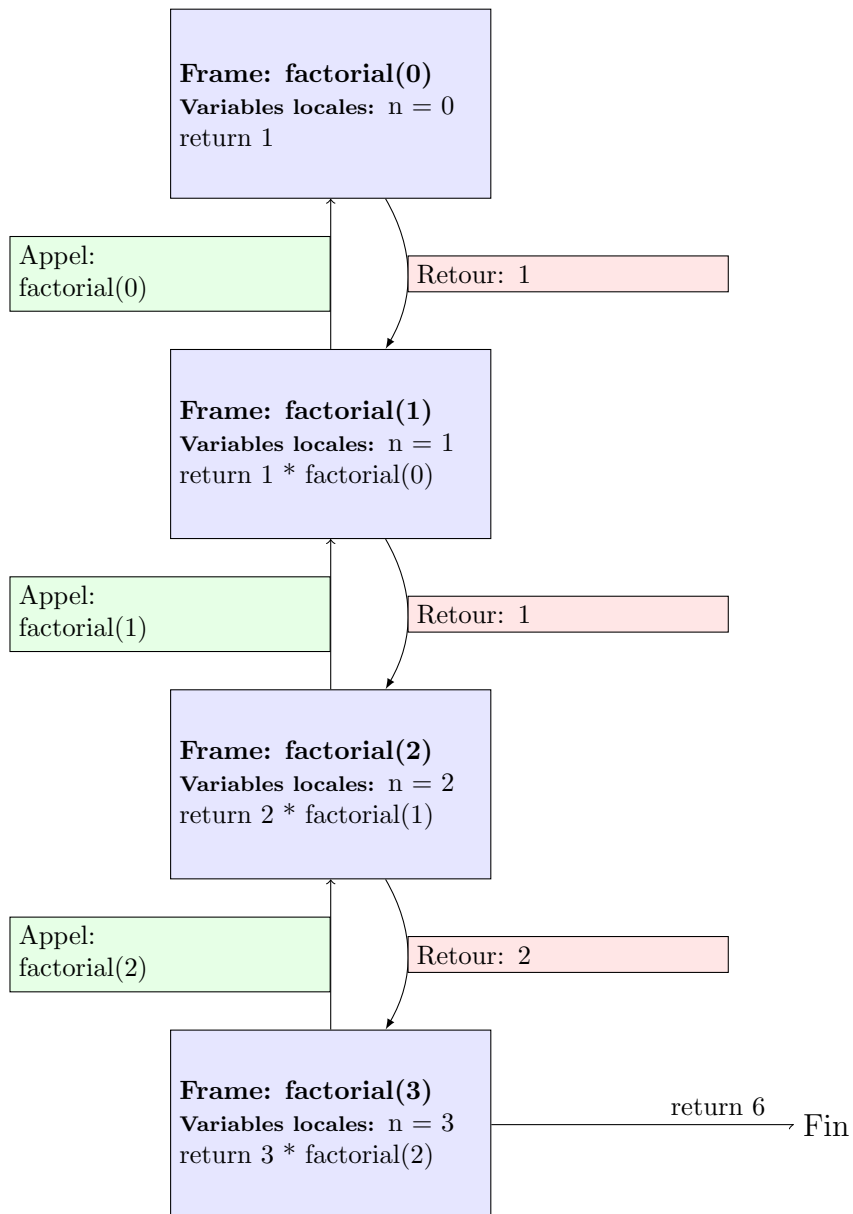
2 Illustration des appels récursifs

Exemple de fonction récursive en Python

```
1 def factorial(n):
2     print(f"{n}, id={id(n)}", sep=" -> ")
3     if n == 0:
4         return 1
5     else:
6         return n * fact(n - 1)
7
8     factorial(3)
```

Exemple: Calcul de Factorielle

- Fonction récursive : `factorial(n)`
- Cas terminal : `factorial(0) = 1`
- Cas récursif : `factorial(n) = n * factorial(n-1)`



Sortie de la fonction fact(3)

```
n=3 -> id= 140601198497608
n=2 -> id= 140601198497576
n=1 -> id= 140601198497544
n=0 -> id= 140601198497512
6
```

Conclusion

- Le premier appel est empilé au bas de la pile, et chaque appel successif est ajouté au-dessus.
- La pile d'exécution fonctionne de bas en haut, où le fond de la pile représente le premier appel.
- Une fois le cas terminal est atteint, les fonctions sont dépilées en sens inverse, avec les valeurs de retour propagées du haut vers le bas.
- Les flèches de retour montrent comment les valeurs sont transmises de chaque appel à son appelant.