



Informatique 1

12. Récursivité

Objectifs du cours

Pouvoir utiliser la récursivité


- ▶ Définir la récursivité
- ▶ Montrer un exemple concret
- ▶ Voir une technique pour écrire des méthodes récursives

Méthode récursive

Une méthode est dite ***récursive*** si elle s'appelle elle-même.

Définition

```
void foo () {  
    ...  
    foo ();  
    ...  
}
```



Exemple factorielle

$$n! = \prod_{i=1}^n = 1 \times 2 \times 3 \dots \times (n-1) \times n \quad \text{and} \quad 0! = 1$$

$$n! = \begin{cases} 1 & \text{if } n = 0 \\ n \times ((n-1)!) & \text{if } n > 0 \end{cases} \quad \forall n \in \mathbb{N}$$

Divide and conquer

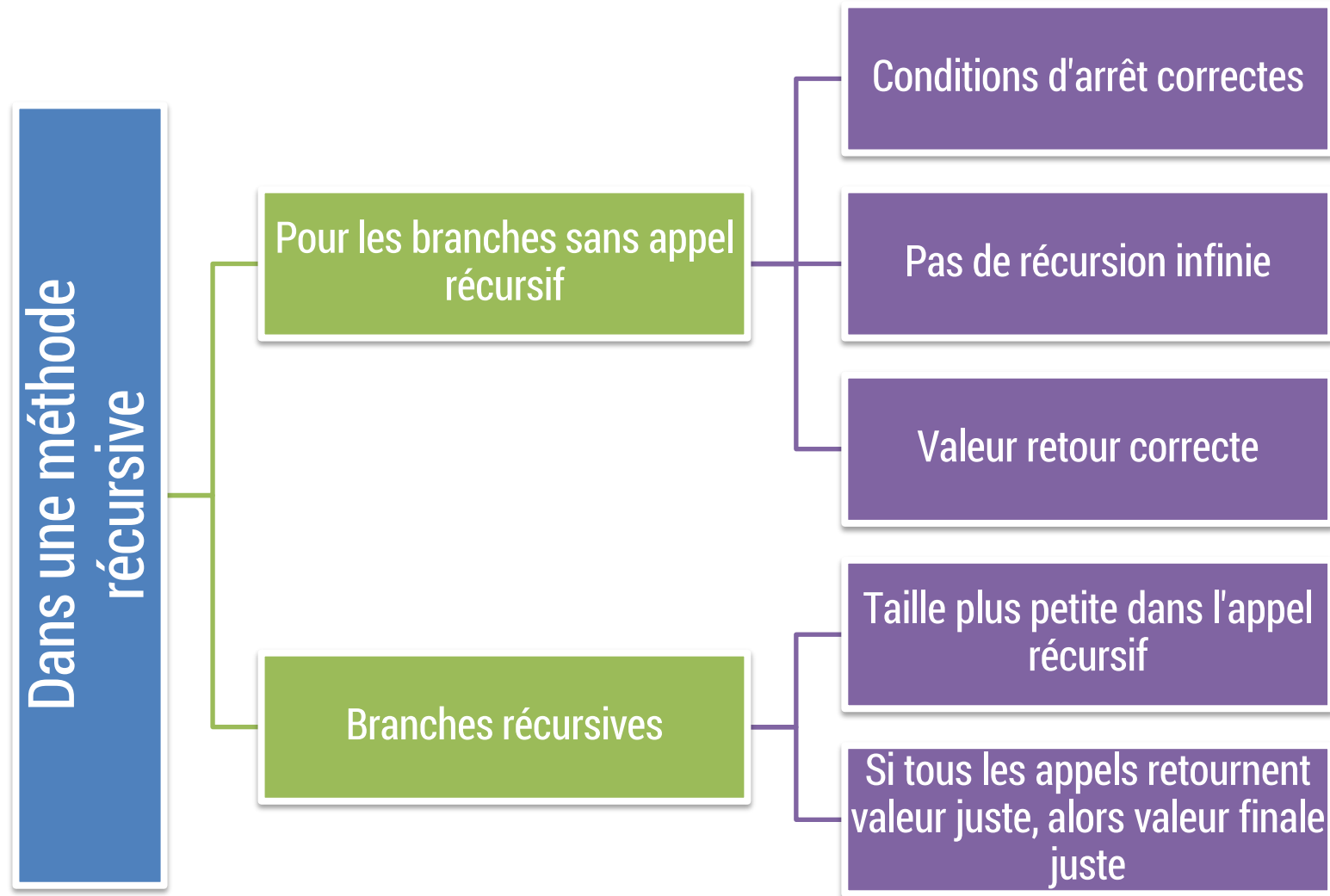
- *Idée*: couper une tâche complexe en sous-tâches plus simples

```
void foo(int n) {  
    ...  
    foo(n-1);  
    ...  
}
```


Vu de près

- Processeur garde la trace ainsi :
 - ▶ Si une méthode réursive est trouvée, exécution de la méthode en cours temporairement stoppée pour l'appel de fonction réursive
 - Car il doit connaître résultat
 - Résultats intermédiaires sauvés
 - ▶ Quand la méthode appelée est terminée, retourne à la méthode arrêtée

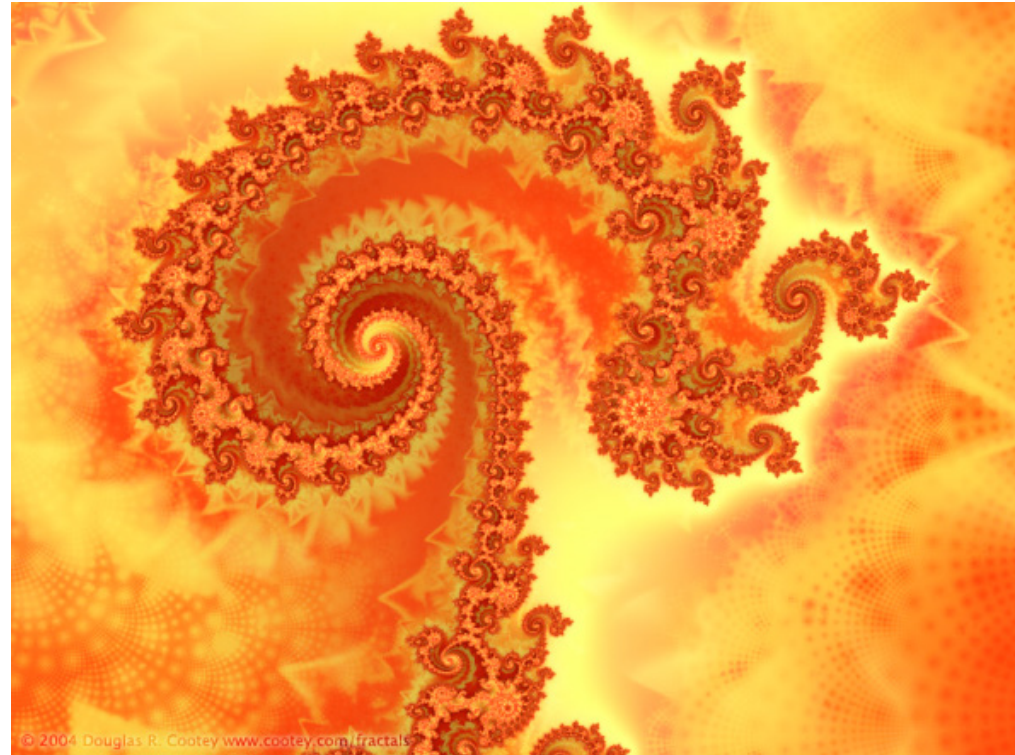
Penser récursivement, recette



Ré recursions dans la nature



En informatique...



Conclusion

- Réursion parfois manière très naturelle d'exprimer du code
- En général difficile, mais toujours possible de dérécurifier

