

Récurtivité

① Exemples

Plan

- 1 Exemples
- 2 Idée générale

Plan

- 1 Exemples
- 2 Idée générale
- 3 syntaxe

Plan

- ① Exemples
- ② Idée générale
- ③ syntaxe
- ④ piège

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de somme_carre(3) :

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de `somme_carre(3)` :

① `somme_carre(3)` ? $3**2 + \text{somme_carre}(2)$

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de `somme_carre(3)` :

- 1 `somme_carre(3)` ? $3**2 + \text{somme_carre}(2)$
- 2 `somme_carre(2)` ? $2**2 + \text{somme_carre}(1)$

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de somme_carre(3) :

- 1 somme_carre(3) ? $3^2 + \text{somme_carre}(2)$
- 2 somme_carre(2) ? $2^2 + \text{somme_carre}(1)$
- 3 somme_carre(1) ? $1^2 + \text{somme_carre}(0)$

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de somme_carre(3) :

- 1 somme_carre(3) ? $3**2 + \text{somme_carre}(2)$
- 2 somme_carre(2) ? $2**2 + \text{somme_carre}(1)$
- 3 somme_carre(1) ? $1**2 + \text{somme_carre}(0)$
- 4 somme_carre(0) ? 0

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de `somme_carre(3)` :

- 1 `somme_carre(3)` ? $3**2 + \text{somme_carre}(2)$
- 2 `somme_carre(2)` ? $2**2 + \text{somme_carre}(1)$
- 3 `somme_carre(1)` ? $1**2 + \text{somme_carre}(0)$
- 4 `somme_carre(0)` ? 0
- 5 `somme_carre(1)` ? $1**2 + 0 = 1$

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de `somme_carre(3)` :

- 1 `somme_carre(3)` ? $3**2 + \text{somme_carre}(2)$
- 2 `somme_carre(2)` ? $2**2 + \text{somme_carre}(1)$
- 3 `somme_carre(1)` ? $1**2 + \text{somme_carre}(0)$
- 4 `somme_carre(0)` ? 0
- 5 `somme_carre(1)` ? $1**2 + 0 = 1$
- 6 `somme_carre(2)` ? $2**2 + 1 = 5$

Exemple : une fonction récursive

```
def somme_carre(n) :  
    if n==0 :  
        return 0  
    else :  
        return n**2 + somme_carre(n-1)
```

Calcul de `somme_carre(3)` :

- 1 `somme_carre(3)` ? $3**2 + \text{somme_carre}(2)$
- 2 `somme_carre(2)` ? $2**2 + \text{somme_carre}(1)$
- 3 `somme_carre(1)` ? $1**2 + \text{somme_carre}(0)$
- 4 `somme_carre(0)` ? 0
- 5 `somme_carre(1)` ? $1**2 + 0 = 1$
- 6 `somme_carre(2)` ? $2**2 + 1 = 5$
- 7 `somme_carre(3)` ? $3**2 + 5 = 14$

Exemples : objet défini de manière récursive

Une liste d'entiers est soit :

- 1 la liste vide []

Autre exemple : un entier naturel est soit :

- 1 0

Exemples : objet défini de manière récursive

Une liste d'entiers est soit :

- 1 la liste vide []
- 2 une liste d'entiers suivie d'un entier

Autre exemple : un entier naturel est soit :

- 1 0
- 2 le successeur d'un entier.

Idée générale

Les douze travaux d'Astérix : la maison des fous.

Fonctions récursives définies sur les entiers naturels

Soit $f : \mathbb{N} \rightarrow E$ une fonction. elle est définie de manière récursive si :

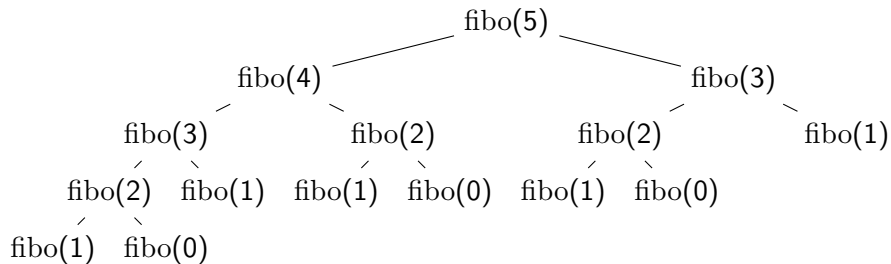
- 1 on connaît la valeur de $f(0)$
- 2 pour tout entier $n \in \mathbb{N}$, on connaît $f(n)$ en fonction de $f(n-1), \dots, f(0)$.

Il est possible de définir des fonctions récursives sur d'autres objets, par exemple les listes. On procède alors de la même façon : on définit la valeur de f pour la liste vide, et on donne la valeur d'une liste de taille n en fonction d'une liste de taille $n-1$.

```
def fonction_recursive(n) :  
    if n==0 :  
        return valeur  
    else :  
        return formule en fonction de fonction_recursive(n-1)
```

```
def fibo(n) :  
    if (n==0) or (n==1) :  
        return 1  
    else :  
        return fibo(n-1) + fibo(n-2)
```

Exemple : calcul de `fibonacci(5)`.



Bonus : quelques idées de projets

- 1 codes correcteurs d'erreur

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données
- ③ algorithmique du texte (lien avec l'alignement des séquences ou autre)

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données
- ③ algorithmique du texte (lien avec l'alignement des séquences ou autre)
- ④ le problème des mariages stables (ou comment les écoles recrutent)

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données
- ③ algorithmique du texte (lien avec l'alignement des séquences ou autre)
- ④ le problème des mariages stables (ou comment les écoles recrutent)
- ⑤ les problèmes de rendu de monnaie

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données
- ③ algorithmique du texte (lien avec l'alignement des séquences ou autre)
- ④ le problème des mariages stables (ou comment les écoles recrutent)
- ⑤ les problèmes de rendu de monnaie
- ⑥ multiplication rapide de polynômes

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données
- ③ algorithmique du texte (lien avec l'alignement des séquences ou autre)
- ④ le problème des mariages stables (ou comment les écoles recrutent)
- ⑤ les problèmes de rendu de monnaie
- ⑥ multiplication rapide de polynômes
- ⑦ problème de coloriage de graphes (le théorème des quatre couleurs)

Bonus : quelques idées de projets

- ① codes correcteurs d'erreur
- ② compression de données
- ③ algorithmique du texte (lien avec l'alignement des séquences ou autre)
- ④ le problème des mariages stables (ou comment les écoles recrutent)
- ⑤ les problèmes de rendu de monnaie
- ⑥ multiplication rapide de polynômes
- ⑦ problème de coloriage de graphes (le théorème des quatre couleurs)
- ⑧ modélisation d'une épidémie et stratégies

Bonus : quelques idées de projets

- 1 codes correcteurs d'erreur
- 2 compression de données
- 3 algorithmique du texte (lien avec l'alignement des séquences ou autre)
- 4 le problème des mariages stables (ou comment les écoles recrutent)
- 5 les problèmes de rendu de monnaie
- 6 multiplication rapide de polynômes
- 7 problème de coloriage de graphes (le théorème des quatre couleurs)
- 8 modélisation d'une épidémie et stratégies
- 9 algèbre linéaire : analyse en composantes principales