

A retenir du chapitre 11 : Récursivité



- Une fonction récursive est une fonction s'appelant elle-même.
- Une fonction récursive doit atteindre une condition d'arrêt qui est souvent un cas de base.
- Une fonction récursive peut nécessiter un nombre d'appels récursifs trop élevé et donc avoir un coût très important.
- Fonction d'exponentiation rapide (prend comme arguments x et n et renvoie x^n) :

```
def exprap(x,n) :  
    if n==0 :  
        return 1  
    else :  
        y=exprap(x,n//2)  
        if n%2==0 :  
            return y*y  
        else :  
            return x*y*y
```

A retenir du chapitre 11 : Récursivité



- Une fonction récursive est une fonction s'appelant elle-même.
- Une fonction récursive doit atteindre une condition d'arrêt qui est souvent un cas de base.
- Une fonction récursive peut nécessiter un nombre d'appels récursifs trop élevé et donc avoir un coût très important.
- Fonction d'exponentiation rapide (prend comme arguments x et n et renvoie x^n) :

```
def exprap(x,n) :  
    if n==0 :  
        return 1  
    else :  
        y=exprap(x,n//2)  
        if n%2==0 :  
            return y*y  
        else :  
            return x*y*y
```