

## Introduction à la programmation dynamique

**Exercice 1 : (Suite de Fibonacci)**

Implémentez en Python un programme qui calcule les  $n$  premiers nombres de la suite de Fibonacci :

$$\begin{cases} f(0) & = 1 \\ f(1) & = 1 \\ f(n+1) & = f(n) + f(n-1) \end{cases}$$

1. Dans un premier temps, vous implémenterez une version récursive.
2. Dans un deuxième temps, vous implémenterez une version basée sur la programmation dynamique.
3. Enfin, vous implémenterez une version itérative sans utiliser de tableau.

Observez que la complexité des deux implémentations est vraiment différente.

**Exercice 2 :**

Implémentez en Python un programme pour résoudre le problème du sac à dos. On donne en entrée une liste d'objets auxquels sont associés à la fois un poids et un bénéfice. Le problème du sac à dos est de choisir les objets à emporter en maximisant la somme des bénéfices sans pour autant dépasser un poids total donné aussi en entrée.

1. Quelle structure de données proposez-vous pour représenter les données d'entrée.
2. Comment stocker les données intermédiaires construites au cours de l'algorithme ?
3. Implémentez l'algorithme qui construit la solution à tous les sous-problèmes de manière itérative : il doit remplir toutes les valeurs de  $B(w, k)$  pour tout entier  $k \leq N$  et pour tout entier  $w < W$ .
4. Implémentez un algorithme complètement récursif.
5. Implémentez l'algorithme de programmation dynamique (appels récursifs, mais en mémorisant les résultats intermédiaires).
6. Implémentez ensuite un algorithme prenant en entrée le résultat de l'algorithme de programmation dynamique ainsi que la structure de données stockant les résultats intermédiaires et qui calcule une liste d'objets à prendre pour obtenir une solution optimale.
7. Modifiez ensuite l'algorithme itératif de manière à avoir la liste de toutes les listes d'objets permettant d'obtenir l'optimum.

On utilisera les données suivantes et on cherchera l'optimum pour un poids de 12 :

objet	0	1	2	3	4	5
poids	5	2	4	6	3	1
bénéfice	14	6	13	17	10	4