TP 4 : Dictionnaires et programmation dynamique (partie 1)

1 Nombre de 1 consécutifs

Le but de cet exercice est de déterminer quel est le plus grand nombre de 1 consécutifs dans une liste `L` donnée.

1. On note `D[i]` le nombre de 1 consécutifs dans `L` en partir du rang `i`. Par exemple, pour `L=[3,2,1,1,2,1,4,1,1,1]`, on a `D[8]=2`.

2. Donné une expression de `D[i]` selon la valeur de `L[i]`.

3. En utilisant un dictionnaire `D` et la programmation dynamique, déterminer le plus grand nombre de 1 consécutifs dans une liste `L`. Vérifier que vous obtenez bien 3 sur l'exemple précédent.

2 The rod cutting problem

On considère une barre d'acier de longueur n fixe, à couper sur mesure pour la revente. Pour chaque longueur, le prix de revente est connu.

longueur i	0	1	2	3	4	5	6	7	8	9	10
prix	0	1	5	8	9	10	17	17	20	24	26

Le but est de maximiser le gain. Dans tout ce problème, on note gain(n) la valeur optimale du gain partant d'une barre de longueur n.

1. On se place dans le cas d'une barre de longueur 4. Envisager toutes les manières de couper la barre (ou de ne pas la couper). Déterminer une solution optimale en termes de gain. Même question pour une barre de longueur 10. *Vous devez trouver* 27 *comme gain optimal.*

2. Créer une liste `prix` tel que `prix[i]` corresponde au prix de revente d'une barre de longueur `i`.

Pour résoudre ce problème, on peut découper un morceau de taille i, situé à gauche de la barre de longueur et obtenir un reste de longueur $n - i$. On recoupe ensuite uniquement ce reste.

3. Donner une relation simple entre gain(n), gain(n-i) et prix(i) pour $1 \leqslant i \leqslant n$.

4. (De haut en bas, version récursive naïve) Écrire une version récursive `gain1` qui à partir d'un entier `n` représentant la longueur initiale de la barre, retourne la valeur optimale du gain. *Vérifier que* `gain1(10)` *retourne* 27.

5. Que retourne `gain1(25)` ? `gain1(50)` ? On peut montrer que la complexité de cet algorithme est exponentielle.

6. (De haut en bas, version récursive avec mémoïsation) Écrire de même la fonction `gain2` qui renvoie la valeur optimale du gain en effectuant une mémoïsation à l'aide d'un dictionnaire. *Vérifier que* `gain2(50)` *retourne* `141`.

On prend le problème à l'envers et on abandonne l'idée de récursivité. On va ici déterminer le gain optimal pour une barre de longueur j de manière itérative en utilisant les gains optimaux préalablement calculés pour des barres de longueur $i \in [\![1, j-1]\!]$.

7. a) Exprimer le gain gain(j) en fonction des prix d'une barre de longueur i et de gain(0), gain(1),..., gain($j-1$).
   b) (De bas en haut) Écrire une fonction itérative `gain3` qui renvoie la valeur optimale du gain. *Vérifier votre fonction avec* `gain3(10)` *et* `gain3(50)`.

8. Reconstruction de la solution optimale à partir de l'information calculée.
   a) Écrire, en modifiant la fonction `gain3`, une fonction `solution` d'argument n entier qui renvoie une liste contenant les valeurs optimales du gain pour tous les entiers j inférieurs ou égaux à n ainsi qu'une liste contenant les longueurs de la première barre à découper pour optimiser le gain pour ces mêmes barres.
   Vérifier que `solution(10)` *retourne* `([0, 1, 5, 8, 10, 13, 17, 18, 22, 25, 27], [0, 1, 2, 3, 2, 2, 6, 1, 2, 3, 2])`.
   b) Écrire une fonction `decoupage` d'argument n entier qui utilise la fonction `solution` et qui retourne la liste des découpes à effectuer pour maximiser le gain à partir d'une barre de longueur n.
   Vérifier que `decoupage(10)` *retourne* `[2,2,6]`.

3 Décomposition d'un entier en parties distinctes

On considère la fonction f, qui à deux entiers a et b ($b \geqslant 1$), associe le nombre de décompositions de a comme somme d'entiers distincts de $[\![1, b]\!]$. Par exemple, $f(10, 5) = 3$ correspond aux 3 décompositions :

$$10 = 5 + 4 + 1 = 5 + 3 + 2 = 4 + 3 + 2 + 1$$

On convient que $f(a, b) = 0$ si $a < 0$.

1. Que vaut $f(a, 1)$ lorsque $a \geqslant 1$?

2. On suppose $a \geqslant 1$ et $b \geqslant 2$. En distinguant les décompositions qui contiennent l'entier b des autres, donner une expression de $f(a, b)$ à l'aide de plusieurs $f(\cdots, b-1)$.

3. Quelle valeur donner à $f(0, b)$ pour que la relation ci-avant soit cohérente ?

4. Écrire une fonction récursive `f1` qui retourne $f(a, b)$ à partir de deux arguments entiers `a` et `b`. Vérifier `f1(10,5)=3` et `f1(100,100)=444793`.

5. Écrire une version modifiée avec mémoïsation `f2` de la fonction `f1`. Vérifier $f(1000, 100) =$ 63317267236458610841 3.