

# DS 1 : dictionnaire et programmation dynamique

Rappel : si une question demande la complexité, il faut la justifier (brièvement si c'est évident).

## 1) Fonctions partielles

On représente une fonction  $f : I \rightarrow \mathbb{N}$  (où  $I \subset \mathbb{N}$ ) par un dictionnaire  $f$  dont les clés sont les entiers  $n$  pour lesquels  $f(n)$  est défini et les valeurs sont les entiers  $f(n)$ .

Par exemple, si  $I = \{1, 2, 4\}$  et  $f(1) = 5, f(2) = 0, f(4) = 9$  alors  $f = \{1: 5, 2: 0, 4: 9\}$ .

1. Écrire une fonction  $\text{sup}(f)$  renvoyant  $\max_{n \in I} f(n)$ , où  $I$  est l'ensemble des clés de  $f$ .
2. Écrire une fonction  $\text{injective}(f)$  déterminant si  $f$  est injective, si possible en complexité linéaire en la taille de  $f$  (c'est-à-dire son nombre de clés).
3. Écrire une fonction  $\text{inverse}(f)$  renvoyant la fonction  $f^{-1}$  (sous forme de dictionnaire) si  $f$  est injective. Si  $f$  n'est pas injective, la fonction renvoie `None` (`return None`).

## 2) Nombre de partitions

Soit  $E$  un ensemble. Une partition  $\mathcal{P}$  de  $E$  est un ensemble de parties de  $E$  deux à deux disjointes dont l'union est  $E$ . Autrement dit :

- $\forall A \in \mathcal{P}, A \neq \emptyset$
- $\forall A, B \in \mathcal{P}, A \neq B \Rightarrow A \cap B = \emptyset$
- $\bigcup_{A \in \mathcal{P}} A = E$

Par exemple, si  $E = \{1, 2, 3\}$  alors  $\mathcal{P} = \{\{1, 2\}, \{3\}\}$  est une partition de  $E$ .

On note  $p_{n,k}$  le nombre de partitions de  $\{1, 2, \dots, n\}$  à  $k$  parties. Par exemple,  $p_{3,2} = 3$  car il y a trois partitions de  $\{1, 2, 3\}$  à deux parties :  $\{\{1, 2\}, \{3\}\}, \{\{1, 3\}, \{2\}\}$  et  $\{\{1\}, \{2, 3\}\}$ .

1. Que vaut  $p_{n,1}$  ?  $p_{n,n}$  ?
2. Montrer que  $p_{n,k} = p_{n-1,k-1} + kp_{n-1,k}$ .
3. Écrire une fonction récursive simple  $p(n, k)$  qui renvoie  $p_{n,k}$ . On n'utilisera pas de programmation dynamique ni de mémorisation.
4. Expliquer brièvement pourquoi la fonction  $p(n, k)$  est inefficace. On ne demande pas de calculer sa complexité.
5. Écrire une fonction  $p\_dyn(n, k)$  qui renvoie  $p_{n,k}$  en utilisant la programmation dynamique. On utilisera une matrice pour stocker les résultats intermédiaires.
6. Donner la complexité de la fonction  $p\_dyn(n, k)$ .