Conservatoire National des Arts et Métiers
EPN 5 – Département Informatique Année 2026–2027

UE RCP104 – Optimisation en informatique

Feuille d'exercices no 2 : programmation dynamique

Exercice 1 : algorithme de programmation dynamique pour le problème du sac à dos, et notion de borne supérieure

On considère 9 objets, chacun muni d'une valeur v_i et d'un poids p_i. Le tableau ci-dessous donne les valeurs de v_i et p_i pour chaque i :

Numéro de l'objet (i)	**Valeur de l'objet** i (v_i)	**Poids de l'objet** i (p_i)
1	15	2
2	6	5
3	9	3
4	3	4
5	2	5
6	1	7
7	20	6
8	30	10
9	40	12

On cherche à sélectionner un ensemble d'objets de valeur totale maximum et dont le poids ne dépasse pas le poids maximum supporté par le sac, qu'on supposera ici égal à $P_{\max} = 18$. Pour cela, on appliquera l'algorithme de programmation dynamique adéquat vu en cours, puis on précisera bien la valeur optimale obtenue, et l'ensemble d'objets correspondant. Comme il y a 9 objets, et que le second membre $P_{\max}$ vaut 18, le tableau à remplir pour exécuter cet algorithme sera composé de 9 lignes et 19 colonnes.

Expliquer ensuite pourquoi la valeur optimale ainsi obtenue est une borne supérieure sur la valeur optimale qu'on obtiendrait en ajoutant la contrainte qu'il n'est pas possible de sélectionner plus que p objets, pour une certaine valeur p donnée (par exemple, $p = 1$, ou bien $p = 3$, ou même $p = 9$, etc.).

Exercice 2 : les 2 variantes de l'algorithme de programmation dynamique pour le problème du sac à dos

On considère 5 objets, chacun muni d'une valeur v_i et d'un poids p_i. Le tableau ci-dessous donne les valeurs de v_i et p_i pour chaque i :

Numéro de l'objet (i)	**Valeur de l'objet** i (v_i)	**Poids de l'objet** i (p_i)
1	2	1
2	10	4
3	1	1
4	2	2
5	4	12

On cherche à sélectionner un ensemble d'objets de valeur totale maximum et dont le poids ne dépasse pas le poids maximum supporté par le sac, qu'on supposera ici égal à $P_{\max} = 15$. Pour cela, on appliquera l'algorithme de programmation dynamique adéquat vu en cours, puis on précisera bien la valeur optimale obtenue, et l'ensemble d'objets correspondant. Comme il y a 5 objets, et que le second membre $P_{\max}$ vaut 15, le tableau à remplir pour exécuter cet algorithme sera composé de 5 lignes et 16 colonnes.

On appliquera ensuite la seconde variante de cet algorithme de programmation dynamique, dans laquelle, pour chaque valeur seuil possible $V \geq 0$ et pour chaque numéro d'objet $i \geq 1$, on calcule le poids minimum d'un ensemble d'objets (choisis parmi les i premiers) dont la valeur totale dépasse (ou est égale à) cette valeur seuil V.