

# Les algorithmes heuristiques

ECE - Outils pour la logistique

2016-2017

## Exercice 1 — *Un problème d'ordonancement*

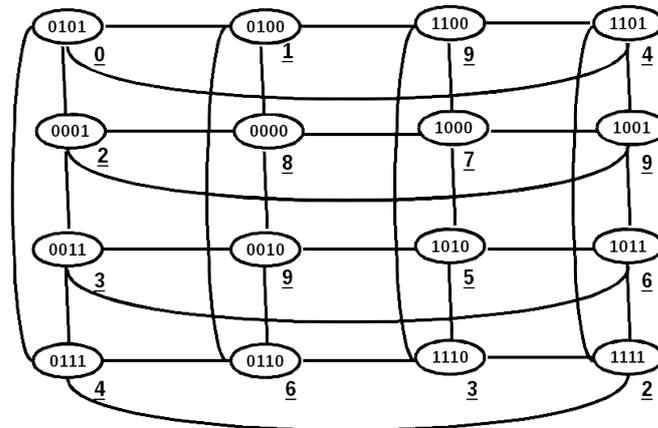
Un contrat a été signé pour un projet informatique à réaliser dans une durée maximale de 13 mois. Ce projet a été décomposé en tâches confiées à des ingénieurs. Ces sept tâches ont entre elles des contraintes de précédence représentées par le tableau ci-dessous. Ce tableau indique également la durée prévue des tâches ainsi que le nombre d'ingénieurs qui doivent être affectés à chaque tâche.

Tâche	Durée (en mois)	Tâches préalables	Nombre d'ingénieurs
A	3	aucune	8
B	8	A	4
C	5	A	4
D	1	C,G	2
E	4	D	6
F	2	C	6
G	6	aucune	4

1. En faisant abstraction du nombre d'ingénieurs disponibles, donner la durée minimale du projet dans le meilleur des cas en appliquant la méthode MPM. Indiquer les dates de début au plus tôt et au plus tard de chacune des tâches, ainsi que leurs marges totales et libres. Préciser ensuite les tâches critiques.
2. En réalité, la société ne dispose que de 12 ingénieurs pour ce projet. On va comparer deux méthodes (heuristiques) différentes pour ordonnancer les tâches au mieux (chacune d'elles permettant de prendre en compte les contraintes de ressources) : dans la première méthode, on donne la priorité à la tâche disponible de plus petite date au plus tard, et, dans la seconde, on commence par placer chacune des tâches critiques de telle sorte qu'elle débute à sa date au plus tôt, puis on applique la première méthode sur les tâches restantes. Pour chaque méthode, illustrer la solution obtenue à l'aide d'un diagramme dit "de GANTT" (avec en abscisse le temps, et en ordonnée le nombre d'ingénieurs). L'une de ces deux méthodes permet-elle d'obtenir une solution respectant le délai maximale du contrat (13 mois) ?

## Exercice 2 — *Heuristique Tabou*

Dans cet exercice, nous considérons les 16 premiers entiers écrits en binaires. Ces vecteurs de tailles 4 représentent l'ensemble des solutions réalisables d'un problème de minimisation. On considère comme voisinage d'une solution  $x$  le vecteur  $x'$  tel que la distance de hamming entre  $x$  et  $x'$  est de 1. En d'autres termes on peut passer d'une solution  $x$  à une solution  $x'$  si  $x$  et  $x'$  ne diffèrent que d'un bit. On obtient le graphe suivant où les arcs représentent les voisinages accessibles pour chaque solution (sommet) :



Pour chaque solution la valeur de fonction objectif associée est indiquée en souligné. Le but de l'exercice est de parcourir l'ensemble des solutions réalisables au cours d'un algorithme heuristique Tabou où la taille de la liste des solutions taboues varie. Nous partons de la solution (1, 1, 1, 1) de valeur 2 et le but est d'atteindre la solution optimale (0, 1, 0, 1) de valeur 0 en le plus petit nombre d'itération.

1. Exécutez l'algorithme avec une liste taboue de taille 1. Atteignez vous la solution optimale ?
2. Exécutez l'algorithme avec une liste taboue de taille 2. Atteignez vous la solution optimale ?
3. Exécutez l'algorithme avec une liste taboue de taille 3. Atteignez vous la solution optimale ?
4. Exécutez l'algorithme avec une liste taboue de taille 4. Atteignez vous la solution optimale ?