UNIVERSITÉ GRENOBLE ALPES **M1-info**

COMPLEXITÉ **2015-16**

Examen du 10 mai 2016

Règles générales

- Durée : 3 heures.
- Seul document autorisé : une feuille A4 recto-verso.
- Toutes les réponses doivent être justifiées.
- Toutes les questions valent autant. On pourra obtenir le maximum en faisant l'équivalent de 15 questions sur 18 parfaitement.
- Il est inutile et même contre-indiqué de recopier l'énoncé sur la copie d'examen.

Énoncé

Complexité de programmes

On rappelle que pour toute paire d'entiers n et p ($p > 0$),[1] il existe un seul entier k et un seul entier r tels que $r < p$ et $n = k \times p + r$.
On note alors :
- $k = n \texttt{ div } p$ (k est le résultat de la division entière de n par p).
- $r = n \texttt{ mod } p$ (r est le reste de la division entière de n par p, appelé le modulo de n par p).

Soit le programme pseudo-Pascal $Prog(i, x)$ suivant, qui prend en entrée deux entiers i et x :

```
Prog(i, x) :
e1 :   y := x;
e2 :  for j := 0 to i do
e3 :     res := y mod 2; y := y div 2
e4 :  end
e5 :  return res;
```

1. Quel et le résultat de $Prog(0, 5)$? Donner également le résultat de $Prog(1, 5)$, $Prog(2, 5)$ et $Prog(3, 5)$.
2. Quelle fonction f calcule ce programme ?
3. Evaluer le nombre d'opérations arithmétiques, d'affectations, et de tests réalisé par lors de l'exécution du programme en fonction de x et de i. En déduire l'ordre de grandeur (Θ) de la complexité du programme en fonction de i.
4. Donner le *principe* d'une machine de Turing classique (à une bande) calculant f selon la même méthode algorithmique à partir de la configuration intiale $q_0|^{i+1}B|^{x+1}$.
5. Donner l'ordre de grandeur de la complexité de cette machine de Turing en fonction de la taille de l'entrée, c'est-à-dire en fonction de $i + x$.

---

1. Si $p = 0$ (division par zéro), on peut étendre la définition usuelle en convenant que que $r = n$, et que $k = 0$, pour avoir l'unicité.

Réduction entre problèmes

Le problème **TS** du voyageur de commerce *(Traveller Salesman)* se définit de la façon suivante : étant donnés

— un ensemble fini V de n villes (numérotées de 1 à n),
— une matrice d de distances telle que pour tout $i \in [1..n]$ et pour tout $j \in [1..n]$ $d(i, j)$ est un entier qui représente la distance en kilomètres entre les villes i et j,
— une constante b (entière),

existe-t-il une *tournée* de l'ensemble des villes dont la *longueur* est inférieure ou égale à b ?

Une *tournée* de n villes (numérotées de 1 à n), est un n-uplet $(p_1, \ldots, p_n)$ formé d'entiers de 1 à n tous différents.

La *longueur* d'une tournée $(p_1, \ldots, p_n)$ est la somme des distances pour visiter les villes de la tournée dans l'ordre et revenir à la première ville, soit : $(\Sigma_{i=1}^{i=n-1} d(p_i, p_{i+1})) + d(p_n, p_1)$

1. Montrer par récurrence que le nombre de tournées d'un ensemble de n villes est égal à $n!$
[Rappel : $0! = 1$ et $(n + 1)! = n! \times (n + 1) = 1 \times 2 \ldots n \times (n + 1)$].

2. Écrire un programme pseudo-Pascal qui prenant en entrée une matrice d de distances, une constante b et une tournée $(p_1, \ldots, p_n)$ renvoie comme résultat $Vrai$ si la longueur de la tournée est inférieure ou égale à b, et $Faux$ sinon. Quelle est l'ordre de grandeur de sa complexité en fonction du nombre n de villes ?

3. Rappeler la définition de l'ensemble NP et montrer que le problème TS est dans NP.

4. Le problème **HC** du circuit hamiltonien *(Hamiltonian Circuit)* se définit sur l'ensemble des graphes de la façon suivante : étant donné un graphe (orienté) $G = (S, E)$, où S est un ensemble de n sommets (numérotés de 1 à n) et où E est l'ensemble de ses arêtes ($E \subseteq S \times S$), existe-t-il un *circuit* contenant chaque sommet du graphe une et une seule fois ?

   Un *circuit* de G est une séquence $(s_1, \ldots, s_k)$ de nœuds de S, chacun (sauf le dernier qui est relié au premier) étant relié au suivant par une arête, formant ainsi un chemin fermé dans le graphe, c'est-à-dire : $(\forall j \in [1..k - 1])$ [$(s_j, s_{j+1}) \in E$ et $(s_k, s_1) \in E$].

   Dessiner un graphe à 5 sommets qui est une instance positive de HC, ainsi qu'un graphe à 5 sommets qui est une instance négative de HC.

5. Soit la transformation f qui, à partir d'un graphe quelconque $G = (S, E)$ construit une instance (V, d, b) de TS de la façon suivante, où n est le nombre de sommets de G ($n = |S|$).
   — $V = S$;
   — pour tout $i \in [1..n]$ et pour tout $j \in [1..n]$: si $(i, j) \in E$, alors $d(i, j) = 1$ sinon $d(i, j) = 2$;
   — $b = n$.

   Donner l'ordre de grandeur de la complexité de la construction $f(G)$ en fonction du nombre n de nœuds du graphe, puis en fonction de la taille t du graphe.

6. Montrer que G est une instance positive de HC si et seulement si $f(G)$ est une instance positive de TS.

7. Dire si on peut en déduire que $HC \preceq_p TS$ ou que $TS \preceq_p HC$. Justifier précisément votre réponse.

Max2SAT est NP-complet

Le problème $Max2SAT$ est défini de la façon suivante : étant donnés F une conjonction de clauses dont chaque clause a au plus 2 littéraux, et k un entier, $Max2SAT(F, k)$ est vrai si et seulement si il existe une interprétation dans laquelle au moins k clauses de F sont évaluées à $Vrai$.

1. Montrer que le problème $Max2SAT$ est dans NP.
2. Soit $F : E_1 \wedge \ldots \wedge E_i \ldots \wedge E_m$ une conjonction de clauses telle que chaque clause E_i a exactement 3 littéraux. On construit à partir de F la formule $f(F)$ en remplaçant chaque clause E_i de F de la forme $\alpha \vee \beta \vee \gamma$ par la conjonction de clauses suivante, dénotée $f_i(\alpha \vee \beta \vee \gamma)$, où w_i est une nouvelle variable propositionnelle et où α', β', γ' sont respectivements les conjugués de α, β, γ :

   $$\boxed{\begin{gathered} f_i(\alpha \vee \beta \vee \gamma) = \\ (\alpha) \wedge (\beta) \wedge (\gamma) \wedge (w_i) \wedge (\alpha' \vee \beta') \wedge (\beta' \vee \gamma') \wedge (\alpha' \vee \gamma') \wedge (\alpha \vee \neg w_i) \wedge (\beta \vee \neg w_i) \wedge (\gamma \vee \neg w_i). \end{gathered}}$$

   On rappelle que deux littéraux h et h' sont dits *conjugués* l'un de l'autre s'il existe une variable propositionnelle a telle que $[h = a \texttt{ et } h' = \neg a]$ `ou` $[h = \neg a \texttt{ et } h' = a]$.

   Donner le résultat de la transformation $f(F')$, où F' est la conjonction de 2 clauses suivante :

   $$F' = (\neg p_1 \vee p_2 \vee p_3) \wedge (p_1 \vee p_2 \vee \neg p_3)$$

   Vous détaillerez $f_1(E_1)$ et $f_2(E_2)$, où E_1 et et E_2 sont les clauses de F' : $E_1 = (\neg p_1 \vee p_2 \vee p_3)$ et $E_2 = (p_1 \vee p_2 \vee \neg p_3)$.
3. Montrer que toute interprétation I dans laquelle $\alpha \vee \beta \vee \gamma$ est évaluée à $Vrai$ peut être étendue à la nouvelle variable w_i pour obtenir une interprétation I_i dans laquelle exactement 7 clauses de $f_i(\alpha \vee \beta \vee \gamma)$ sont évaluées à $Vrai$.
4. Montrer que, dans toute interprétation I' dans laquelle $\alpha \vee \beta \vee \gamma$ est évaluée à $Faux$, au plus 6 clauses de $f_i(\alpha \vee \beta \vee \gamma)$ sont évaluées à $Vrai$.
5. En déduire que, pour toute conjonction F de m clauses ayant exactement 3 littéraux chacune, F est satisfaisable si et seulement si il existe une interprétation dans laquelle au moins $7 \times m$ clauses de $f(F)$ sont évaluées à $Vrai$.
6. On rappelle que le problème $3SAT$ (défini sur l'ensemble des conjonctions de clauses ayant exactement 3 littéraux chacune), consistant à déterminer s'il existe une interprétation dans laquelle toutes les clauses sont évaluées à $Vrai$, est NP-complet.

   Rappeler la définition d'un problème NP-complet et démontrer que $Max2SAT$ est NP-complet.