

Commentaires DS2

Exercice 1 : Problèmes NP Complet

La méthode à connaître parfaitement (pour savoir ce qui doit être démontré ...)

Cycle-Ham NP : détailler un peu la vérification (sans y passer trop de temps s'il n'y a pas de difficulté)

Réduction : ne pas se tromper de sens ...

Donner un nom aux instances :

(G,s,t) instance (i, quelconque !) de Chaîne-Ham

G' = instance (transformée de i) de Cycle Ham

Démontrer :

$\Rightarrow$: si (G,s,t) admet une chaîne-ham, alors G' admet un cycle-ham

$\Leftarrow$: si G' admet un cycle-ham, alors (G,s,t) admet une chaîne-ham

- transformation attendue : voir corrigé

- des transformations qui ne fonctionnent pas :

- relier s et t par une arête

- ajouter un sommet relié à tous les autres

- ...

Ne pas oublier de conclure

Problème :

Le sujet était faisable en temps imparti

Attention aux confusions C / OCaml

Ne pas forcément chercher à optimiser une fonction plus qu'il ne le faut (surtout si le code devient plus long à écrire et à expliquer ...)

Q12 et Q23 permettent de distinguer les candidats les plus à l'aise en programmation.

Q1 : respecter le formalisme : $X^* = (0,0,1,1,0)$: un vecteur de 0 / 1

Q3 : $X_{opt} = X$ ne fonctionne pas en C

Q4 : c'est +1 en binaire : algorithme de Droite à Gauche (propagation de la retenue)

Q6 : $2^N \times N$: nombre de vecteurs différents (par dénombrement) $\times$ coût de traitement d'un vecteur

Q7 : backtracking : essayer de faire simple. donner clairement la spécification de la question

Q8 : $2^N \times N$: à établir soit en détaillant l'arbre des appels, soit par récurrence : $C(n) = 2 * C(n-1)$ et $C(0) = N$

Q9 : élagage facile

Q11 - Q12 : énoncé pas toujours bien compris. on attendait un élagage un peu moins facile.

En C : la division de deux entiers renvoie un entier

Approches gloutonnes

Q13 : tri sélection, avec une fonction de comparaison. Si ordre = 1, il n'y a rien à faire.

ATTENTION : erreur dans le corrigé : $i < N-1$ (optim)

$j < N$

Q14 : algo glouton. on attend un calcul incrémental du poids (optim facile qui évite une complexité quadratique)

Q15 : l'algo glouton examine tous les objets (ne s'arrête pas au premier objet non sélectionné). La stratégie 3 donne la solution optimale sur l'exemple : il faut donner un vrai contre-exemple

Q16 - Q17 - Q18 : on attend une réponse en une phrase

Q19 : c'est le point délicat de la programmation dynamique : trouver la bonne relation de récurrence et la justifier correctement.

Q20 : le tableau peut permettre de valider l'idée de la Q19

Q22 : reconstruction de la solution à partir du tableau (classique assez difficile)

Q23 : la question de programmation la plus difficile du sujet

Q24 - Q25 : la complexité en $N \cdot W$ (remplissage du tableau) n'est pas linéaire en la taille de l'entrée (la taille de W est $\log_2 W$) ; donc non on n'a pas démontré que $P = NP$...