

Informatique

Présentation du sujet

Le sujet est composé de deux parties indépendantes.

La première partie s'attache à résoudre le problème du voyageur de commerce. Après quelques considérations pratiques de programmation (I.A), elle étudie la complexité théorique du problème (I.B), les limites théoriques (NP-complétude, absence d'epsilon-approximation dans le cas général). Enfin, elle demande de justifier théoriquement l'algorithme de Christofides et de l'implémenter en langage C.

Cette première partie évalue plusieurs compétences théoriques des candidats : la réduction d'un problème à un autre, la NP-complétude, les graphes... En outre, elle évalue la compréhension des concepts de programmation en langage C (pointeurs, allocation dynamique de mémoire notamment).

La seconde partie, plus abstraite, étudie des graphes dont les sommets sont eux-mêmes des arbres. Les arêtes de ces graphes sont définies via des transformations sur les arbres. En plus de cet aspect théorique, cette partie demande aux candidats de programmer en OCaml.

Analyse globale des résultats

Le sujet comporte 52 questions. Les candidats ont, en moyenne, traité 25 questions. Les questions impliquant de la programmation ont été plébiscitées par les candidats.

Dans l'ensemble, les candidats ont démontré de bonnes compétences en programmation à la fois en langage C et en OCaml, montrant qu'ils pouvaient écrire des programmes complexes. Les résultats aux questions théoriques sont plus variables, quelques candidats les évitant. Le jury est globalement satisfait du niveau atteint par les candidats.

Les questions de cours ont donné des résultats mitigés.

Les candidats savent rédiger des preuves et, pour la plupart, savent faire preuve de concision. La concision dans la rédaction était explicitement demandée dans certaines questions, les candidats l'ont, pour la plupart, bien respectée, allant à l'essentiel. On a constaté, dans les copies de cette année, un lien direct entre le manque de concision et les erreurs dans les preuves.

Commentaires sur les réponses apportées et conseils aux futurs candidats

Les candidats maîtrisent globalement bien les langages C et OCaml. La syntaxe n'est pas un problème pour une large majorité des candidats. Un conseil qui peut être donné aux candidats est de bien vérifier la spécification des fonctions qu'ils programment. Leur fonction doit la satisfaire entièrement. Sur certaines copies, seule une partie de la spécification a été respectée.

Le sujet portait sur les graphes et demandait de construire des réductions, de construire des graphes, d'écrire des preuves... Il est assez facile d'illustrer tout cela, par des schémas. De trop nombreux candidats ont préféré tout décrire sous forme de texte là où un schéma aurait rendu le discours plus clair. Le jury encourage fortement les candidats à faire un usage plus fréquent des schémas.

Le sujet demande d'écrire de nombreuses preuves. Si la plupart des candidats savent écrire des preuves, un certain nombre d'erreurs ont mené des candidats à prétendre prouver des assertions fausses. La preuve est un moyen de vérifier si un résultat est vrai, ou s'il ne l'est pas. L'objectif n'est pas de pouvoir tout prouver, mais bien de ne pouvoir prouver que ce qui est vrai. La preuve est censée présenter une certaine

« résistance ». Le jury encourage les candidats à se demander, lorsqu'ils écrivent une preuve « Aurais-je échoué à écrire ma preuve si le résultat que je souhaite prouver était faux ? ».

Quelques remarques sur certaines questions

Q4. La première partie de cette question est une question de cours : il s'agit de transformer un problème d'optimisation en un problème de décision. La majorité des candidats (60%) a bien traité ce point.

Dans la deuxième partie de la question, il faut montrer que le problème de décision ainsi obtenu est NP. Pour cela, il ne faut rien oublier, notamment que la taille du certificat doit être polynomiale.

Q5, Q6 et Q7. Ces questions demandent de réduire un problème de graphe à un autre problème de graphe. Il est utile dans ces circonstances de faire un schéma pour expliquer la transformation que l'on souhaite faire. Lorsque un candidat cherche à démontrer que sa transformation est correcte, l'objectif n'est pas de paraphraser la construction, mais de vérifier que la construction est bien correcte. Un certain nombre de copies ont donné des « preuves » qu'une construction incorrecte était juste.

Q8. Dans cette question, il est attendu des candidats qu'ils fassent preuve d'esprit de synthèse et donnent les idées directrices plutôt que de lister tous les cas particuliers. Ce que les étudiants ont fait dans leur majorité : 86 % du total des candidats ont traité correctement cette question et ont obtenu le maximum des points ; 94 % des candidats ont tenté de traiter cette question et 6 % ne l'ont pas traitée.

Q16. La question 16 demande d'implémenter un algorithme difficile du programme de MPI, l'algorithme de Kruskal, en un temps assez limité. Les résultats sont mitigés, les candidats qui ont traité cette question ont, en moyenne, obtenu à peu près la moitié des points correspondants.

Q18. Cette question n'a pas posé de réelle difficulté aux candidats, 79 % d'entre eux ont traité parfaitement cette question et obtenu la totalité des points.

Q19. Cette question présentait plusieurs difficultés en même temps : déterminer la taille du tableau à créer, allouer dynamiquement un tableau, traiter le résultat du calcul (le nombre de sommets doit être écrit au travers d'un pointeur, tandis que le pointeur vers le tableau doit être renvoyé). 87 % des candidats ont traité cette question et, parmi ceux qui l'ont traité, 98 % ont donné un résultat au moins partiellement juste, mais seulement 35 % ont donné un résultat totalement juste.

Quand une question comme celle-ci présente plusieurs points à traiter, il faut les traiter tous, et ne pas en négliger certains.

Q20. Cette question n'a été traitée que par un candidat sur quatre. Elle ne présente pourtant pas de difficulté majeure.

Q31 à Q37. Ces questions sur les arbres exigeaient, pour certaines, des dessins, des schémas. Les meilleures copies sont celles qui ont utilisé des schémas y compris dans les questions où ce n'était pas obligatoire.

Q41 à Q49. Ces questions demandant de programmer en OCaml ont été bien réussies par les candidats qui les ont traitées.

Conclusion

Le jury est satisfait du niveau global des copies. La partie pratique est le point fort des candidats, malgré la difficulté qu'il peut y avoir à écrire du code sur une feuille de papier.

Les plus grandes différences entre candidats portent sur les aspects théoriques, sur la maîtrise de la notion de preuve, sur la capacité à écrire des preuves, sur la capacité à concevoir un algorithme, sur la compréhension des questions plus abstraites, comme, dans ce sujet, les réductions de problèmes.