TSI

CONCOURS COMMUN INP
RAPPORT DE L'ÉPREUVE ÉCRITE
INFORMATIQUE
Session 2024

1/ PRÉSENTATION DU SUJET

COLORER UN GRAPHE

Ce sujet abordait la thématique des graphes, une nouveauté du programme d'informatique.

Partie I - Des algorithmes pour colorer un graphe

Trois algorithmes gloutons étaient étudiés, chacun pouvant être vu comme un raffinement du précédent.
Le but était d'utiliser le moins de couleurs possibles pour colorer un graphe.
Le premier, un algorithme intuitif de coloration.
Le second algorithme étudié (Welsh-Powell) permettait de mettre en place un algorithme de tri.
Le troisième algorithme (DSATUR) était nettement plus difficile à implémenter.
Enfin, un petit détour par la notion de « clique » terminait cette partie.

Partie II - Interrogation d'une base de données géographiques

Les questions étaient très classiques :

- une requête simple ;
- une requête avec jointure ;
- une agrégation ;
- une question plus difficile : plusieurs approches étaient possibles.

2/ REMARQUES GÉNÉRALES

Erreurs les plus fréquentes

Les correcteurs sont conscients que le nombre d'heures d'enseignement dédiées à l'informatique n'est pas très important en classe préparatoire. Il en découle chez un certain nombre d'étudiants une maîtrise parfois approximative du langage. Aussi, dans la mesure du possible, une erreur dans la connaissance du langage Python n'est pénalisée qu'une seule fois dans la copie. Les démarches globalement justes et les initiatives prises sont valorisées.
Il est conseillé d'attacher une grande importance à la nature des objets manipulés (ceux reçus en arguments et ceux renvoyés par les fonctions) : des entiers, des listes, des listes de listes, etc.

Il y a eu parfois des confusions lors des parcours de listes : celles-ci peuvent être parcourues soit par indices (`for i in range(len(L))`) soit par élément (`for elt in L`).
On note aussi des `return True/return False` mal placés dans les fonctions.
Il est remarqué une difficulté chez certains étudiants à admettre et/ou réutiliser les questions précédentes pour progresser dans le sujet.
Certains candidats ont mal géré leur temps. Ils ont commencé par les questions difficiles de la première partie et ils n'ont pas abordé les questions simples de la seconde partie sur les bases de données.

Remarques sur le sujet, le texte et sa compréhension

L'énoncé a été jugé clair et de difficulté croissante. Il portait sur une partie nouvelle du programme mais les étudiants ont plutôt bien réagi.

Problèmes ou difficultés rencontrés lors de la correction

Le sujet pouvant être considéré comme long pour un étudiant « standard » de TSI, le barème a été défini en conséquence.
Les étudiants sérieux de la filière pouvaient ainsi obtenir une note très honorable sans aborder la totalité du sujet.

Bilan

Cette épreuve a permis aux étudiants de TSI de montrer leurs compétences acquises au cours des deux années de préparation.
Des progrès restent à faire sur la notion de complexité. Mentionner de « simples boucles », de « boucles imbriquées » et de « boucles successives » suffit souvent à donner des réponses pertinentes à ces questions. Les correcteurs souhaitent bien sûr que la complexité trouvée soit en cohérence avec le code proposé par le candidat !

3/ REMARQUES SPÉCIFIQUES

Partie I - Des algorithmes pour colorer un graphe

Q1. Bien traitée par de nombreux candidats.

Q2. Bien traitée par de nombreux candidats.

Q3. Question explicitement au programme qui s'est avérée difficilement traitée.
Des réponses en termes de complexité étaient attendues. Quelle est la structure de données qui permet le test d'adjacence le plus rapide ? Quelle est celle qui fournit le plus vite la liste des voisins ? Quid de l'encombrement mémoire ? Quid de la facilité à ajouter ou supprimer un sommet du graphe etc.

Q4. Bien traitée par de nombreux candidats.

Q5. Assez bien traitée par de nombreux candidats, assez souvent sous forme de boucle. Attention à la position des « return » : au sein de la boucle, on quitte la fonction dès que l'on a constaté l'adjacence entre deux sommets. Par contre, c'est après être sorti de la boucle que l'on peut conclure à la non-adjacence.
Bien sûr la syntaxe « `j in LA[i]` » rappelée en annexe simplifiait la tâche…

Q6. Des maladresses mais le parcours des sommets est compris. Même problème qu'en Q5 parfois sur les « `return True/return False` ».
Quelques confusions entre parcours d'une liste par éléments ou par indices.

Q7. La plupart des réponses ont été satisfaisantes notamment chez les candidats ayant écrit une double boucle à la question précédente.

Q8. Bien traitée par de nombreux candidats.

Q9. De nombreux candidats ont oublié de gérer les doublons dans les couleurs des voisins.

Q10. Bien traitée.

Q11. Bien traitée.

Q12. L'erreur la plus fréquente a été de donner la liste des couleurs, non pas dans l'ordre des sommets 0,1,2,…,n-1, mais dans l'ordre de coloration des sommets (la liste des couleurs obtenue était donc inversée).

Q13. Question bien traitée quand elle a été comprise. Quelques étudiants font une boucle pour compter le nombre d'éléments d'une liste, « `len(L)` » irait plus vite !

Q14. Question classique mais pas toujours bien traitée. Les copies de référence n'ont pas été pénalisées cette année.

Q15. Question qui réutilisait Q13 et Q14.

Q16. Beaucoup de réponses possibles, ce qui était attendu était de savoir renverser une liste. Parfois, des erreurs dans la gestion des indices.

Q17. Il fallait faire la synthèse des questions précédentes. Certains candidats réécrivent tout au lieu d'utiliser les fonctions « `ranger` » et « `renverse` » !

Q18. Question de complexité mal traitée ; là aussi les correcteurs attendaient une cohérence avec le code proposé.

Q19. De bonnes réponses pour le code, un peu moins pour la complexité.

Q20. Question assez mal traitée.

Q21. La difficulté était de renvoyer le nombre de couleurs différentes utilisées pour les voisins, pas le nombre de voisins colorés.

Q22. Question souvent bien traitée quand elle a été abordée.

Q23. Question souvent bien traitée quand elle a été abordée.

Q24. Question difficile et donc très sélective.

Q25. Il y a eu un nombre significatif de bonnes réponses pour la première partie de la question.

Q26. Peu traitée.

Q27. Peu traitée, les réponses partielles pertinentes ont été valorisées.

Partie II - Interrogation d'une base de données géographiques

Q28. Bien traitée.

Q29. Il s'agissait de faire une jointure, cette technique n'est pas toujours maîtrisée.

Q30. La fonction « `SUM` » n'est pas toujours connue. Oubli assez fréquent de « `or code_pays2 ='F'` ».

Q31. Question difficile. Quelques bonnes réponses cependant. Les démarches partiellement correctes ont été valorisées.

4/ CONCLUSION

Les correcteurs ont pu constater une bonne maîtrise de la manipulation des graphes par liste d'adjacence chez de nombreux candidats.

Certains candidats ont montré une belle finesse dans la manipulation des objets et des concepts.
Par contre, d'autres ont été pénalisés par leur trop faible maîtrise des bases de Python. Les correcteurs sont convaincus qu'un étudiant assidu et impliqué durant les deux années de préparation est capable d'obtenir une note (au moins) très honorable à cette épreuve.