Épreuve écrite d'Algorithmique et Informatique

Il y avait cette année 162 candidats à l'épreuve. Les notes obtenues vont de 2 à 20, avec une moyenne de 10 et un écart-type de 3,5.

Description de l'épreuve
L'épreuve, d'une durée de 45 minutes, consiste en des questions autour du chiffrement de Vigenère, du déchiffrement et de la cryptanalyse par analyse fréquentielle. De niveau progressif, le sujet débute par quelques questions préliminaires sur le typage de données, quelques QCM (identifier la fonction correcte parmi 4 proposées) puis se poursuit par la programmation ou la modification de fonctions nécessaires et se termine par un code à compléter pour déterminer la clé de chiffrement et décrypter un texte chiffré avec la méthode de Vigenère. Les différentes parties du programme ont été abordées.

Analyse de l'épreuve

1. Questions préliminaires
1.a et 1.b : Questions bien réussies dans l'ensemble : la question a été réussie par 80% des candidats. Les types de variables semblent bien assimilés par la plupart des candidats, certains ont décalé d'un rang (1 pour A, 2 pour B, au lieu de 0 pour A, etc.)
D'autre part, certains candidats confondent encore un type informatique (type chaine de caractère, type entier, …) et un ensemble mathématique (ex : les entiers naturels).

2. Chiffrement de Vigenère

2.a i et ii : 90% des candidats ont su comprendre et expliquer correctement la fonction proposée. De nombreux candidats ont en plus su remarquer que l'énoncé comportait ici une coquille. Le jury a été indulgent sur la notation de la question, accordant des points à toutes les réponses cohérentes.

2.b : Il s'agissait de corriger une fonction qui comportait 2 erreurs : rajouter « range » dans la boucle for (ce qui n'a pas était vu par la moitié des candidats) et rajouter des [ ] ou utiliser append pour ajouter un élément dans une liste (ce qui n'a pas était également vu par la moitié des candidats). Au final,1/3 des candidats a eu le total des points , 1/3 la moitié des points et 1/3 zéro point.

2.c. d. e. : Il s'agissait de 3 QCM où il fallait choisir l'unique fonction qui réalisait l'objectif demandé. Les bons candidats répondent correctement aux 3 questions. Trop de candidats se sont laissés piéger par des choses simples : indentation qui faisait que l'instruction n'était plus dans la boucle (la moitié des candidats s'était laissée piéger), la somme de 2 listes qui avait été confondue avec une somme terme à terme, alors qu'il s'agit de la concaténation des 2 listes (la moitié des candidats n'avait pas trouvé la bonne réponse).

Au total, ces questions très simples (parties 1 et 2) auraient pu rapporter 9 points à tous les candidats.

3. Déchiffrement
Cette partie consistait à écrire deux fonctions décrites dans l'énoncé.

3.1 La première fonction consistait à faire la différence terme à terme de 2 listes d'entiers, modulo 26 (la fonction modulo était rappelée dans l'énoncé). Question plutôt bien réussie : seulement 40% des candidats ont eu moins de la moitié des points ou juste la moitié à cette question simple.
3.2 La seconde fonction consistait à adapter une fonction qui était donnée dans une question précédente. Assez peu de réponses totalement correctes : 80% des candidats ont eu moins de la moitié des points ou juste la moitié à cette question. Les candidats n'ont pas tous remarqué qu'il suffisait d'adopter le code donné en 2.e

4. Cryptanalyse par analyse fréquentielle
4.a il s'agissait à nouveau d'un QCM. Il est demandé de trouver la fonction qui calculait correctement l'indice de coïncidence de 2 chaines de caractères. Le principe de l'indice de coïncidence était rappelé juste au-dessus. Une des questions les mieux réussies : les 2/3 des candidats ont trouvé la bonne fonction.

4.b Il s'agissait d'écrire une fonction utilisant l'indice de coïncidence calculé en 4.a pour réaliser un test de Friedman (expliqué dans l'énoncé). Il s'agissait de tester tous les entiers successivement pour décaler la chaine de caractère et de s'arrêter quand on avait trouvé un entier dont l'indice de coïncidence était supérieur à 0.07. Il fallait donc faire une boucle while et un compteur ou bien une boucle for dont on sort sous condition.
Les ¾ des candidats n'ont pas traité la question. Parmi ceux qui l'ont traité, seul un tiers l'a réussi avec le maximum de points ou presque.

4.c Il s'agissait de corriger une fonction donnée dans l'énoncée, notamment en y incluant un calcul de maximum. Alors que le calcul de maximum est au programme seuls 4 ou 5 candidats ont complètement réussi la question. Cette question au demeurant n'a été traitée que par la moitié des candidats.

4.d et 4.e : il s'agissait d'assembler les fonctions utilisées dans les différentes parties de l'épreuve pour réaliser le décryptage d'un message codé. Plus de 90% des candidats n'ont traité aucune de ces 2 questions.
A l'issue de la péréquation, le barème de l'épreuve a finalement été constitué de manière à ce que les questions 4c, 4d et 4e soient grosso-modo uniquement des points bonus (épreuve notée sur 27)

Conclusion :
Au final, le niveau des copies est plutôt faible et assez décevant par rapport à l'an dernier. Beaucoup de points étaient pourtant faciles à prendre.
On note qu'il y a moins de confusion sur le typage, mais il reste des difficultés avérées vers la fin du sujet pour modifier un code dans un but donné.
Comme toujours il y a heureusement des candidats qui se distinguent et réussissent à avoir de très bonnes notes, mais l'histogramme montre qu'ils sont très peu nombreux, notamment si on le compare à l'histogramme de l'année dernière.
La relative baisse de niveau par rapport à l'année dernière peut s'expliquer par le fait que l'épreuve orale de mathématiques ne comporte plus d'Informatique, et que les professeurs de mathématiques y consacrent donc raisonnablement moins de temps.