

Composition d'Informatique (2 heures), Filières PSI et PT

Rapport de M. Dominique ROSSIN, correcteur.

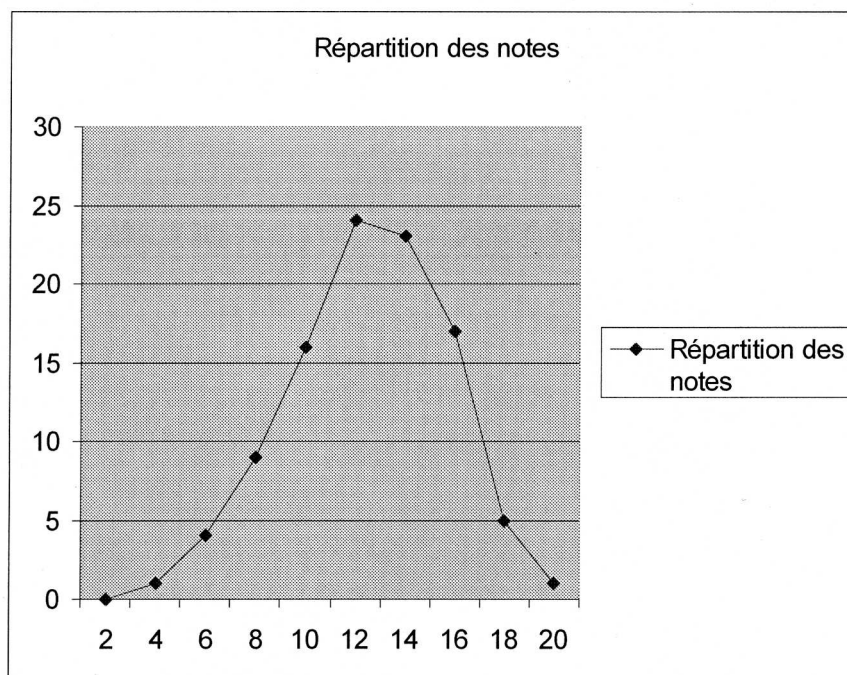
Les statistiques et le bilan donnés ci-après concernent uniquement les candidats admissibles.

Bilan

La répartition des notes pour les 99 candidats français admissibles est la suivante :

$N < 5$	$5 \leq N < 9$	$9 \leq N < 13$	$13 \leq N < 17$	$17 \leq N$
3 %	27 %	45 %	24 %	1 %

La moyenne des copies s'élève à 10.7 avec un écart-type de 3.2. Une analyse plus fine des notes donne la distribution suivante :



Le langage de programmation utilisé est majoritairement Maple (que trop de candidats

continuent à orthographier Mapple). Néanmoins, certains candidats ont choisi Java, Mathematica ou encore Pascal comme l'indique le tableau suivant :

Pascal	Mathematica	Maple	Java
5	4	90	1

Commentaires

Toutes les questions du sujet demandaient l'écriture d'un programme. Le critère principal d'évaluation des questions est la conformité du programme aux spécifications imposées par l'énoncé, que ce soit du point de vue de la forme que du fond (algorithmes employés, complexité en temps et en espace). La majorité des candidats ne font plus d'erreurs de syntaxe. La quasi-totalité des erreurs est due à des erreurs dans les algorithmes employés ou encore plus généralement à des réponses erronées. Seul l'emploi pas toujours maîtrisé de la structure Maple

for <expr> from <expr> to <expr> while <expr> do <statement> od ;

bien que déjà fortement déconseillée les années précédentes a conduit certains candidats à faire des erreurs au niveau de la terminaison et de la complexité des algorithmes.

Le problème a été bien traité jusqu'à la question 8 incluse par une grande partie des candidats. Seules les questions 5 et 7 ainsi que la fin de l'énoncé ont présenté des difficultés.

Certains langages de programmation ne permettent pas de modifier certains paramètres d'entrée. Bien entendu, cela a été pris en compte lors de la correction et l'intégralité des points a été accordée si les variables avaient la bonne valeur en fin de fonction.

Commentaires détaillés

Les résultats par question ainsi que le barème sont donnés ci-dessous :

Question	1	2	3	4	5	6	7	8	9	10	11
Barème	1	1	1	2	3	1	3	1	2	2	3
Réussite	93%	88%	49%	60%	32%	89%	23%	76%	3%	19%	1%
Moyenne /10	9,74	9,64	7,04	6,92	5,67	9,08	5,84	8,29	2,84	3,38	0,58
Zéro	1%	0%	3%	16%	13%	7%	10%	7%	44%	49%	87%

Question 1

Il s'agissait ici de calculer la somme de certains éléments du tableau. Seul 7% des candidats n'ont pas obtenu la note maximale à cette question. Certains des candidats recopient dans un tableau annexe les valeurs à sommer avant de réaliser l'opération à

l'aide de la fonction ad-hoc de Maple au lieu de réaliser la somme sur le tableau de départ. Une erreur fréquente est aussi le fait de confondre affichage et retour. La bonne solution consistait à parcourir le tableau de départ et pour chaque indice, ajouter à une variable la valeur du chiffre d'affaires de l'entreprise correspondante si celle-ci appartenait à l'alliance donnée en paramètre.

Question 2

Cette question consistait à vérifier que le résultat de la question précédente était ou non supérieur à une certaine valeur. Il est surprenant de voir que de nombreux candidats réécrivent intégralement la fonction précédemment réalisée dans la question 1.

Question 3 :

Cette question consistait à vérifier que l'alliance donnée était stable. 50% des candidats ont obtenu le maximum de points sur cette question. L'algorithme employé par ces candidats consistait à utiliser la question 1 de manière à calculer la valeur de l'alliance puis à parcourir le tableau des chiffres d'affaires et pour chacun d'eux, *si l'entreprise faisait partie de l'alliance* à vérifier que la somme précédemment calculée diminuée de ce chiffre descendait en dessous de l'objectif. L'erreur la plus courante a été de tester l'intégralité des chiffres d'affaires sans vérifier que l'entreprise considérée faisait partie de l'alliance donnée en paramètre.

Question 4

Il s'agissait ici de remplir l'alliance en commençant pas les entreprises de plus grand chiffre d'affaires, et donc de réaliser une boucle dans le sens décroissant. 60% des candidats ont ainsi obtenu la note maximale à cette question. Certains candidats ont mal compris la notion de minimalité en nombre d'entreprises, conduisant à des programmes erronés et à un grand nombre de 0.

Question 5

Cette question a été assez mal traitée. Il s'agissait ici d'ajouter un à un nombre écrit en binaire. L'énoncé suggérait fortement de réaliser l'opération bit à bit en simulant une addition « à la main » en propageant la retenue. Néanmoins, une majorité de candidats ont préféré calculer le nombre décimal correspondant, avant de faire une incrémentation de 1 et de le reconvertir en binaire. Cette conversion a induit un grand nombre d'erreurs de programmation sans parler des problèmes de taille des entiers qu'une telle opération peut représenter. Cette méthode a ainsi conduit le plus souvent à de mauvaises notes.

Question 6

Cette question a été parfaitement traitée par la quasi-totalité des candidats. La totalité des points a été accordée si le code donné imprimait la liste des entreprises de l'alliance

même si la sortie ne suivait pas au caractère près la sortie proposée dans l'énoncé.

Question 7

Cette question comportait deux parties. La première consistait à recoller ensemble les questions 3, 5 et 6. La clé de la seconde partie était de se souvenir que l'on peut représenter 2^n nombres en base 2 sur n bits. Cette seconde partie a souvent conduit à de nombreuses erreurs ($2n$ ou $2n^2$). Dans la première partie, trop de candidats ont réécrit tout ou partie d'une des questions 3, 5 ou 6, complexifiant inutilement leur code. Cette erreur a été peu pénalisée.

Question 8

Cette question consiste à remarquer qu'une somme partielle de 1 à k est égale à la somme partielle de 1 à $k-1$ augmentée de la valeur de k . Cela conduit naturellement à un algorithme linéaire pour le calcul des sommes partielles. L'erreur la plus courante a consisté en la programmation d'un algorithme en n^2 , recalculant la somme partielle à chaque fois. Cette faute a été lourdement pénalisée.

Question 9

Cette question a posé de nombreux problèmes aux candidats. Il fallait appliquer ce qui était expliqué dans l'énoncé juste avant et rechercher le plus petit entier tel que $t[k']$ soit plus grand que l'objectif partiel. Certains candidats ont interprété cette question comme devant réaliser le processus complet de remplissage. Aucune pénalisation pour ces interprétations n'a été faite tant que la complexité des algorithmes résultants était inchangée.

Question 10

Cette question a été mieux traitée que la précédente. Elle pouvait d'ailleurs être traitée indépendamment. Cette question consistait à chercher la première entreprise plus grande que i n'étant pas dans l'alliance donnée. De nombreuses erreurs ont été commises sur la mise à jour de l'objectif partiel ou lors de la modification de l'alliance courante.

Question 11

Cette question n'a été que très peu traitée par les candidats. Elle consistait à remettre ensemble les questions 8, 9, 10 et 11. Peu de personnes ont trouvé la complexité résultante qui était exprimée en fonction du nombre de solutions.