

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

Rapport de M. Dominique ROSSIN, correcteur.

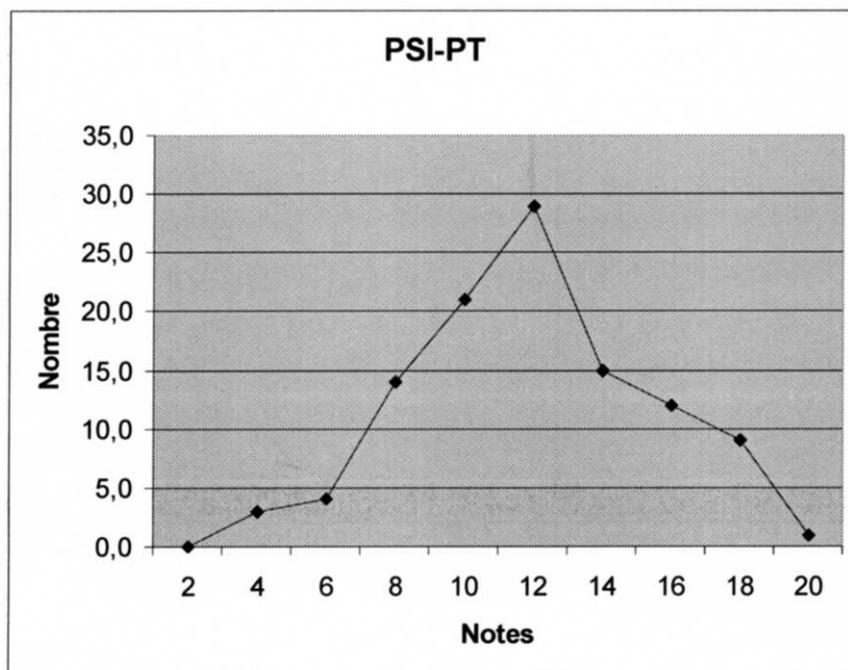
Les statistiques et le bilan donnés ci-après concernent uniquement les candidats admissibles.

I. Bilan

La répartition des notes pour les 108 candidats français admissibles est la suivante :

$N < 5$	$5 \leq N < 9$	$9 \leq N < 13$	$13 \leq N < 16$	$16 \leq N$
4%	24%	45%	18%	9%

La moyenne des copies s'élève à 10.9 avec un écart-type de 3.4. Une analyse plus fine des notes donne la distribution suivante



Le langage de programmation utilisé est majoritairement Maple (Notons une nette amélioration orthographique du nom Maple). Néanmoins, certains candidats ont choisi C, Mathematica ou encore Pascal comme l'indique le tableau suivant :

Pascal	Mathematical	Maple	Java
3	3	101	1

II. Commentaires

De nombreuses questions nécessitaient l'écriture d'un programme. Le critère principal d'évaluation de ces 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).

Un problème que se posent fréquemment les candidats concerne l'impossibilité dans certains langages de programmation de respecter des aspects de l'énoncé comme la numérotation des indices des tableaux qui commence à 0. Une bonne solution employée par la majorité des candidats consiste simplement à spécifier au début de sa copie qu'il a connaissance de ce décalage d'indice dû au langage utilisé mais qu'il considèrera que les indices des tableaux dans son langage commencent à 0 pour respecter l'énoncé.

De même, 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.

Cette année, le niveau des programmes s'est globalement amélioré, les candidats semblent avoir respecté les consignes données l'an dernier en ce qui concerne l'utilisation des structures complexes comme la boucle *for/from/to/while* de Maple et ont préféré dans la majorité des cas utiliser des structures plus simples. Cela permet une meilleure compréhension des programmes par le correcteur et donc une correction plus fine des questions.

Le problème a été bien traité jusqu'à la question 7 puis se produit un seuil entre une grande partie des candidats qui ont bloqué sur cette question et les suivantes et ceux qui ont réussi à franchir ce seuil.

III. Commentaires détaillés

Les résultats par question ainsi que les coefficients respectifs des questions sont donnés ci-dessous :

Question	1	2	3	4	5	6	7	8	9
Barème	1	1	2	1	2	2	2	1	1
Réussite	87%	82%	59%	20%	43%	20%	20%	14%	4%
Moyenne/10	9,65	4,49	8,56	7,57	7,12	5,49	3,66	2,37	0,31
Zéro	0%	1%	2%	5%	7%	18%	50%	61%	93%

Un premier commentaire sur ce tableau concerne la difficulté croissante du sujet avec des moyennes qui décroissent régulièrement. Ensuite, on remarque le seuil de la question 7 où la moitié des candidats sont restés bloqués.

Question 1 et 2 :

Il s'agissait dans ces deux questions de vérifier que les candidats ont compris l'énoncé dans un cas très simple. Cette question a été naturellement très bien traitée.

Question 3 :

Cette question demandait l'écriture d'un programme dans le cas où l'on considère seulement deux types de produits. Certains candidats ont cherché à améliorer la solution proposée en éliminant certains calculs et en considérant que la solution ne pouvait être atteinte que pour les valeurs extrémales du nombre de découpes ce qui était faux.

Question 4 :

Cette question était intéressante dans le sens où l'on cherchait à généraliser la question précédente et certains candidats ont essayé de trouver une méthode efficace pour la résoudre sans tenir compte de la méthode proposée dans l'énoncé. Très souvent, les méthodes proposées se révèlent fausses et les candidats pénalisés. La méthode attendue était de découper le premier objet un certain nombre de fois et ensuite d'appeler la fonction de la question précédente avec les paramètres ad hoc pour calculer les découpes des deux autres objets puis de prendre la meilleure découpe possible.

Question 5 :

Cette question proposait un schéma général pour le calcul d'une solution à base de programmation dynamique bien qu'aucune connaissance dans ce domaine ne soit nécessaire. Il s'agissait de trouver une méthode pour remplir le tableau M en respectant les conditions imposées dans l'énoncé. Beaucoup de candidats ont réussi cette question, à quelques erreurs de programmation près.

Question 6 :

Cette question aurait dû être traitée avec la même réussite que la précédente, les modi-

fications entre les deux étant mineures. Néanmoins, de nombreux candidats ont « oublié » de lire une partie de la question et bien que calculant le tableau D ne réalisait pas l'impression des découpes comme demandé par l'énoncé.

Question 7 :

Cette question demandait de reprendre les questions précédentes mais en limitant la production des produits à une unité. Cette contrainte nécessitait en réalité de reformuler la récurrence énoncée entre la question 4 et la question 5 et ne pouvait se déduire des tableaux précédemment calculés.

Question 8 : Cette question demandait la programmation de la récurrence trouvée à la question précédente. La corrélation entre les deux questions se retrouve d'ailleurs sur la notation. La programmation de cette question ne comportait pas de difficultés majeures et les candidats ayant trouvé la récurrence ont réussi sans difficulté cette question.

Question 9 :

Dans cette question, l'énoncé demandait de reprendre la question précédente en tenant compte de contraintes mémoire. Peu de candidats ont abordés cette question et parmi ceux l'ayant réussi, certains avaient déjà donné cette solution optimale à la question précédente.