

## Composition d'Informatique (2 heures), Filières MP et PC

Rapport de MM. Sylvain CONCHON et Dominique ROSSIN, correcteurs.

Ce rapport et les statistiques incluses dans celui-ci ne prennent en compte que les copies des candidats français.

### I. Le bilan

Rappelons que cette épreuve n'est corrigée que pour les candidats admissibles. Cette épreuve est commune entre les filières MP et PC et cette année, l'écart entre ces deux filières semble s'être considérablement accru.

#### *Filière PC*

Le nombre des candidats français admissibles est de 474 en filière PC. La moyenne de l'épreuve est de 10,97/20 avec un écart-type de 4,09. La note la plus haute est de 20/20 et la note la plus basse est de 0,7/20. Les candidats ont majoritairement traité cette épreuve en répondant aux questions dans l'ordre. Au final, 26% des candidats ont terminé cette épreuve (c'est-à-dire ont tenté de répondre aux 11 questions). Les langages de programmation utilisés sont Maple (93%), Python (3%), Turbo Pascal (2%) et Mathematica (2%). La répartition des notes est la suivante :

$0 \leq N < 4$	20	4,2 %
$4 \leq N < 8$	99	20,9 %
$8 \leq N < 12$	151	31,9 %
$12 \leq N < 16$	147	31,0 %
$16 \leq N \leq 20$	57	12,0 %
Total	474	100 %
Nombre de copies : 474		
Note moyenne : 10,97		
Écart-type : 4,09		

#### *Filière MP*

Le nombre des candidats français admissibles est de 124 en filière MP. La moyenne cette année est très élevée, 13.45/20. Cette moyenne est notamment due au fait que plus de 80% des candidats ont terminé cette épreuve. La répartition des notes est la suivante :

$0 \leq N < 4$	0	0,0%
$4 \leq N < 8$	10	8,1%
$8 \leq N < 12$	21	16,9%
$12 \leq N < 16$	68	54,8%
$16 \leq N \leq 20$	25	20,2%
Total	124	100%
Nombre de copies : 124		
Note moyenne : 13,45		
Écart-type : 3,21		

## II. Commentaires

Le sujet portait sur l'étude de polynômes et leur comparaison autour de 0. Dans un second temps, il s'agissait d'étudier les permutations qui sont des échangeurs de polynômes. Cette année, nous notons une plus grande aisance des candidats à dépasser les contraintes liées au langage.

- Les indices commençant à 1 au lieu de 0.
- L'impossibilité de renvoyer des tableaux comme résultats de fonctions.
- L'impossibilité de déclarer des tableaux dont la longueur dépend d'un paramètre d'une fonction.

Les consignes des années précédentes et de l'énoncé de passer outre ces contraintes de langage ont été respectée par la quasi-totalité des candidats.

La notation des questions de programmation repose avant tout sur la clarté et la bonne formulation du code écrit et bien entendu sur le fait que la solution proposée donne le bon résultat. Selon les questions, une attention particulière a également été portée sur l'« efficacité » du code produit.

## III. Commentaires détaillés

Les différents taux de réussite aux questions sont donnés dans les tableaux ci-après.

### Filière PC

Question	1	2	3	4	5	6	7	8	9	10	11
Réussite %	9	44	61	28	45	11	36	35	56	63	2
Moyenne/10	7,6	7,5	7,8	4,9	6,6	4	5	6,4	6,9	6,7	1,5
Zéro %	3	5	3	39	13	19	38	23	23	30	77

### Filière MP

Question	1	2	3	4	5	6	7	8	9	10	11
Réussite %	18	79	59	54	60	44	42	69	45	88	1
Moyenne/10	5,8	8,6	7,8	7,4	7,3	7,1	7,3	8,6	5,8	9,5	3,3
Zéro %	2	6	2	9	10	6	10	6	5	3	26

**Question 1 :** Il s'agit ici d'une question classique d'évaluer un polynôme. Dans cette question, un bonus a été accordé pour l'utilisation de la méthode de Horner.

**Question 2 :** Le calcul de la valuation d'un polynôme pose le problème de la détection du 0 et donc du test d'arrêt de la boucle qui parcourt les coefficients. Cette question a été assez bien traitée.

**Question 3 :** Le calcul de la différence entre deux polynômes est classique néanmoins, il faut être vigilant quant au degré du résultat.

**Question 4 :** Cette question demandait l'utilisation de la question 1 ainsi que de la question 2. Certains candidats ont toutefois pensé qu'évaluer le polynôme en une valeur proche de 0 (10E-10 par exemple) suffirait. Cette méthode est bien évidemment erronée.

**Question 5 :** Il fallait écrire un algorithme de tri. Un algorithme de complexité quadratique suffisait pour l'obtention de la totalité des points.

**Question 6 :** Cette question simple a fait débat car de nombreux candidats ont pris pour acquis l'existence d'une fonction `verifie_pos` sans l'écrire. Ainsi, nous avons considéré en MP comme bon cette question si la fonction corollaire `verifie_neg` avait bien été programmée auparavant, tandis qu'il fallait donner le code de cette fonction pour avoir tous les points en PC.

**Question 7 et 8 :** Ces questions nécessitait d'imbriquer plusieurs boucles et de réaliser un test. Aucune difficulté notable si ce n'est la taille importante que prenaient certains codes de candidats pour réaliser le test.

**Question 9 :** Cette question demandait de faire très attention à la complexité du calcul. En effet, un calcul récursif donnait un algorithme exponentiel et ne permettait pas d'obtenir le résultat pour des valeurs supérieures à 10.

**Question 10 :** Question très facile pour qui l'abordait. 88% des candidats en MP et 63% en PC l'ont parfaitement réalisé.

**Question 11 :** Cette question a été assez mal traitée. La difficulté résidait dans l'indication de l'énoncé qui demandait d'utiliser la fonction `est_echangeur_aux`. De nombreux candidats n'ont pas compris comment l'utiliser les conduisant à des méthodes erronées. Cette question n'a été abordée que par peu de candidats en PC.