

INFORMATIQUE

Épreuve écrite - filière MP

I) REMARQUES GENERALES

L'épreuve d'Informatique de 1998 est la deuxième épreuve de ce type. La première épreuve, celle de 1997, avait un caractère spéculatif car le niveau des candidats était inconnu. La bonne révélation de 1997 fut le très bon niveau des candidats en programmation, que ce soit en langage CAML ou en langage PASCAL. Cette année la difficulté du sujet a été adaptée au niveau de l'année précédente et donc renforcée sans qu'il en résulte de baisse sensible des résultats.

Dans l'ensemble, le niveau des candidats était très satisfaisant bien qu'inégalement réparti. Comme nous le verrons, des notes similaires mais situées dans la moyenne des notes ne récompensent pas forcément les mêmes qualités.

L'épreuve de 1998 a été bâtie sur le même modèle que celle de 1997 :

- un exercice de logique propositionnelle
- un exercice de théorie des automates finis
- un problème de programmation décliné deux fois: une fois pour le langage PASCAL et une fois pour le langage CAML au choix du candidat.

Les deux sujets de programmation sont identiques. La raison en est que l'on ne veut pas que les candidats choisissent un langage pour leurs études parce qu'ils pensent qu'un des deux sujets était plus facile que l'autre. Cependant, les styles de programmation de PASCAL et CAML sont différents : essentiellement itératif pour le premier, essentiellement récursif pour le second. La nature des problèmes et des structures de données à traiter dans l'un ou l'autre des langages n'est également pas la même. Cet aspect est peut-être à prendre en compte.

II) REMARQUES PARTICULIERES

Certaines copies sont des brouillons. La notation en tient compte. Des candidats n'ont même pas numéroté les réponses aux questions.

Le problème de logique a été moins traité que les autres.

Certains candidats ont résolu les problèmes théoriques et tenté de gagner quelques points en programmation.

Certains candidats ont résolu le problème de programmation et gagné quelques points dans les questions introductives des problèmes théoriques.

La qualité de rédaction des programmes laisse franchement à désirer et il ne s'agit pas là que de la forme, pas du fond.

II-1) Problème de logique

Le plus difficile avec le problème de logique est de trouver un problème assez complexe pour le niveau du concours. Le programme de logique est en effet très restreint. La logique est

une théorie formelle. Dans les logiques plus évoluées que la logique propositionnelle (logiques du premier ordre, classique ou intuitionniste, linéaire ou du second ordre), un grand nombre de résultats fondamentaux sont obtenus par des artifices syntaxiques, donc fondés sur l'utilisation adéquate de notations. Le principal problème rencontré par les candidats semble être un problème de notation car l'objet du problème est normalement connu mais il y a eu beaucoup de copies blanches pour la logique.

Un nombre non négligeable de candidats ont confondu formes normales conjonctives et disjonctives. Il est vrai que le traitement du même problème avec des formes normales disjonctives est plus aisé. Certains ont considéré qu'une forme normale conjonctive ne contenait que des :

Les questions 5, 6 et 7 concernant les circuits logiques ont été particulièrement traitées et avec succès. Une erreur commune, explicable par le stress des candidats, a été la rédaction des tables des fonctions f_1, f_2, \dots, f_7 alors que l'énoncé ne demandait que les tables "des deux fonctions f_1, f_7 ". Beaucoup de travail de rédaction pour rien !

Certains ont écrit les lignes des tables de vérité dans un ordre aléatoire, plutôt que dans un ordre croissant ou décroissant des entrées, rendant la correction fastidieuse sinon impossible.

II-2) Problème de théorie des automates finis

Le graphe FIG. 6 de la question 2 comportait une erreur : la flèche de l'état (3,2,1) vers l'état (2,3,1) était dans le mauvais sens. L'erreur se trouvait dans une question introductive destinée à mettre les candidats dans le bain. Elle était cependant triviale et parfaitement vérifiable par les candidats. De fait, un pourcentage négligeable de candidats (moins de 1 %) ne l'ont pas vue ou pas mentionnée.

Des candidats ignorent ce qu'est un automate déterministe. Certains dessinent des automates visiblement non déterministes avec éventuellement plusieurs états initiaux et démontrent sans hésiter qu'ils sont déterministes.

Cependant, en règle générale, les candidats avaient un bon niveau en théorie des automates finis.

II-3) Problèmes de programmation

Les candidats devaient choisir l'un des deux problèmes uniquement. Les programmes proposés par les élèves ayant choisi PASCAL sont la plupart du temps des programmes itératifs. La récursivité n'est pas tellement utilisée. Ce n'est pas bien grave en PASCAL mais le même phénomène se retrouve en CAML où la récursivité est le moyen normal et naturel de programmer. Les programmeurs CAML ont souvent écrit du PASCAL en syntaxe CAML

Deux petites perles en réponse aux questions 1 des deux problèmes: "on choisit $N=10$ car le décimal est plus facile à traiter par les calculateurs" et "on choisit $N=9$ car ainsi on a le milliard, ce qui est sympathique".

Malgré les recommandations aux candidats leur disant de NE PAS traiter les cas d'erreurs, certains ont cru bon de les traiter avec toutes les complications qui en résultent. Travail inutile !

Beaucoup de candidats n'ont absolument pas tenu compte des retenues lors des additions partielles. Bizarre !

Beaucoup de solutions très compliquées ont été proposées là où des solutions simples s'imposaient.

II-3.1) Problème de programmation en PASCAL

Moins de 20 % des candidats ont choisi le programme de programmation en PASCAL, ce qui témoigne d'une avancée de l'enseignement de CAML au détriment de PASCAL.

Malgré les recommandations très strictes de l'énoncé sur le type **string** (en gras dans l'énoncé), bien des candidats ont traité les chaînes de caractères comme des tableaux de caractères.

Quelques candidats ont utilisé les noms de fonctions comme variables intermédiaires : ce n'est pas rigoureux et ce n'est pas du PASCAL même si certains compilateurs supportent cela.

II-3.2) Problème de programmation en CAML

La notation **prefix** est mal comprise et mal utilisée.

En CAML aussi, certains candidats ont utilisé le type **string** comme un tableau de caractères.

Beaucoup trop de candidats ont programmé de manière itérative. Cela donne des programmes plus compliqués et peu en accord avec l'esprit de CAML.

II-3.3) Remarques

Concernant les exercices de programmation, les candidats doivent savoir que les correcteurs n'attachent pas une valeur essentielle aux problèmes de syntaxe ni à certains détails techniques. Tant qu'il n'y a pas d'abus, il n'y a pas de sanction dans la note. Par exemple, un point-virgule oublié n'est pas pénalisant. Ce qui est principalement jugé est l'algorithme et la connaissance "raisonnable" d'un des deux langages de programmation.