

---

## ÉPREUVE ORALE D'INFORMATIQUE

ENS : PARIS LYON CACHAN

MEMBRES DE JURY : S. HADDAD, E. JEANDEL & O. SERRE

*Coefficients :* PARIS MPI : 20 groupe I : 4  
LYON MPI : 3 groupe I : 4  
CACHAN MPI : 12 groupe I : 12

---

Cette épreuve a concerné les candidats aux trois Écoles Normales Supérieures du concours MP-option informatique et les candidats aux trois Écoles Normales Supérieures du concours informatique. Le jury a examiné 157 candidats (contre 147 l'année dernière). L'épreuve testait en 45 minutes sans préparation l'habileté des candidats à manipuler rigoureusement les objets fondamentaux de l'informatique (automates, langages formels, graphes, logique, combinatoire des mots) et leur demandait une bonne intuition algorithmique. Les notes se sont étalées entre 3 et 20, avec une moyenne de 11,25 et un écart-type de 3,78 (voir l'histogramme en fin de document). Le jury a trouvé que le nombre de candidats très faibles (note inférieure ou égale à 6) a significativement diminué cette année, ce qui est une bonne chose.

Trente-neuf sujets originaux différents ont été proposés aux candidats. Une quinzaine de sujets comportait au moins une question portant sur les langages rationnels ou la reconnaissabilité par automates finis. Il est à noter que les exercices relevant des automates étaient non classiques afin d'évaluer au mieux la capacité du candidat à travailler autour de ce concept. Quelques exercices étudiaient des propriétés simples de combinatoire des mots afin de juger la capacité des candidats à raisonner de façon combinatoire sur des objets simples et familiers. Certains sujets faisaient travailler sur des objets classiques (graphes, algorithmes, variantes des automates) sortant du cadre du programme et dont il fallait dans un premier temps assimiler la définition. Enfin, un nombre important de sujets comportait également un volet algorithmique simple. Il est à souligner que bien que cette épreuve ne soit pas de nature "pratique", il peut être demandé aux candidats d'écrire des algorithmes (parfois en détail) afin de pouvoir les analyser finement.

Le jury souhaite également rappeler que l'épreuve est publique (à condition de l'accord des candidats). On ne saurait trop conseiller aux futurs candidats et à leurs enseignants de venir assister au moins une fois à l'épreuve afin de mieux l'appréhender.

Le jury a particulièrement évalué :

- La capacité d'initiative du candidat, son inventivité, sa manière d'aborder les questions et sa capacité à assimiler rapidement des concepts étrangers.
- La capacité à formaliser un problème et à exprimer mathématiquement les propriétés à prouver.
- La capacité à écrire des preuves propres.
- La capacité à proposer des algorithmes, à en prouver la correction et à analyser précisément la complexité (temporelle, et dans une moindre mesure spatiale).
- La bonne connaissance des notions (et des preuves) au programme.

La plupart des candidats réagissent bien face à des concepts qu'ils ne connaissent pas *a priori*. Les candidats ont pour la plupart de bonnes idées et, même lorsque celles-ci ne conduisent pas au résultat, ils ne se démontent pas, ce que le jury a apprécié. En effet, le jury recherche des candidats pouvant à terme être de bons chercheurs et l'inventivité est l'un des critères fondamentaux. Les

preuves sont souvent écrites proprement, et le jury a été globalement satisfait par les preuves de terminaison et d'analyse de la complexité d'algorithmes. Le jury a particulièrement apprécié que des candidats considèrent des exemples pour mieux appréhender certaines questions.

Les faiblesses les plus marquantes ont été les suivantes :

- Trop de candidats ignorent encore des résultats centraux du cours. En particulier des constructions classiques sur les automates (intersection, détermination par exemple) sont incorrectement données voire tout simplement ignorées. Il est, de notre point de vue, **impossible d'obtenir une note correcte à cette épreuve si l'on est pris en défaut sur un point de cours** : même si le périmètre de l'épreuve est délibérément à la frontière (voir au delà) du programme, les (rares) résultats vus en cours ne peuvent être ignorés des candidats. De plus, un candidat qui méconnaît une construction classique ne pourra s'en inspirer pour en déduire une construction pour un problème original. Ces remarques valent aussi pour le programme de première année. On pense en particulier aux algorithmes de tri, tout candidat devant connaître un algorithme de tri en  $O(n \log n)$ , comme le tri fusion par exemple.
- Certaines preuves et constructions utiles ne sont pas toujours connus sous leurs formes les plus simples ce qui se révèle pénalisant. Pour les automates, il serait bon de connaître une construction directe pour l'intersection plutôt que de se reposer sur l'expression de cette dernière à l'aide du complément et de l'union (faisant ainsi l'économie d'une explosion exponentielle du nombre d'états). Pour prouver qu'il existe des langages non réguliers, il est bon de connaître également la preuve par argument de cardinalité (l'ensemble de tous les langages n'est pas dénombrable, celui des langages rationnel l'étant) car elle se généralise à de nombreuses classes de langages, ce qui n'est pas le cas du lemme de pompage. Sur ce point encore (exemple de langage non régulier), il vaudra mieux considérer un langage comme  $\{a^n b^n \mid n \geq 0\}$  plutôt que le langage  $\{a^{n^2} \mid n \geq 0\}$  sur lequel le lemme de pompage est plus pénible à utiliser.
- De façon surprenante, certains candidats ignorent le nom du théorème de Kleene (ce qui est pénible dans une question du type *rappeler l'énoncé du théorème de Kleene*). De même la notation  $\Sigma^+$  (ensemble des mots finis non vides sur  $\Sigma$ ) est largement méconnue (ce qui n'est pas grave en soit mais qui alourdit souvent les notations dans les raisonnements des candidats). Enfin, il conviendrait que les candidats connaissent les différentes terminologies (équivalentes pour ce qui est des langages de mots finis) : rationnel, régulier, reconnaissable.
- Nous ne saurions que trop encourager les candidats à considérer des exemples pour aborder les questions que nous posons. C'est un réflexe trop rare qui se révèle pourtant payant et qui est à la base du métier de chercheur. De même, rien n'interdit de faire un dessin pour raisonner sur les automates ou, plus encore, sur les problèmes de combinatoire des mots. Bien sûr, cela ne dispensera pas d'une preuve rigoureuse au final.
- Concernant les automates, la notion de non-déterminisme est trop peu exploitée par les candidats, en particulier lorsque l'on souhaite utiliser un automate qui devine / mémorise une information.
- Les candidats se sont révélés particulièrement décevants lorsqu'il s'agissait de faire des raisonnements mathématiques (géométrie euclidienne, dénombrement, opérations sur les ensembles, topologie, quantification logique, etc.) : certes, il s'agit d'une épreuve d'informatique, mais cela n'interdit pas d'utiliser les bases des mathématiques.
- Quelques exercices portaient sur des objets très classiques mais hors programme (les graphes par exemple). Cependant, nous insistons sur le fait que ces objets ne sont pas au programme. Cela ne veut pas dire qu'ils ne peuvent pas être connus des candidats mais ceux-ci doivent alors

maîtriser les concepts qu'ils prétendent connaître sous peine de décevoir le jury. De même, si les élèves invoquent des concepts comme, par exemple, ceux d'automate minimal ou de résiduels, ils devront être capables d'en donner les définitions / constructions / preuves de correction.

- Tous les sujets étaient dactylographiés et fournis dans leur intégralité au début de l'épreuve au candidat. Il pouvait souvent être intéressant de jeter un coup d'œil rapide à l'ensemble des questions, ou tout du moins de lire la question courante en entier avant de l'aborder.

