

Remarques générales

Le sujet couvre les points fondamentaux du programme de l'option d'Informatique : logique, automate, preuve et complexité des algorithmes, programmation en CaML ou Pascal. Il reste ainsi dans la droite ligne des épreuves du concours commun polytechnique des années précédentes. Une petite faiblesse peut être notée dans le cadre de la modélisation en logique qui n'est pas traitée cette année.

La partie I concerne les automates. Elle est semblable aux exercices posés les années 2002 et précédentes : l'étude d'un opérateur de transformation d'automates. Il s'agit cette année de la dérivation d'un automate selon un mot. Cette partie comporte d'une part la manipulation d'un automate particulier et d'autre part une preuve par induction sur les mots (récurrence sur la longueur des mots). Ces deux aspects correspondent à l'essentiel de ce que tous les étudiants doivent être capables de faire en ce qui concerne les automates. Le premier ne pose en général aucun problème. Le second par contre n'est toujours pas traité de manière suffisamment satisfaisante. Le second aspect a donc été beaucoup plus discriminant que le premier.

La partie II concerne l'algorithmique et la programmation dans le cadre particulier de la logique des propositions : il s'agit d'étudier une procédure de complétion d'une base de connaissances représentée par une proposition logique. Cette partie permet d'évaluer quatre éléments du programme : logique, preuve, complexité et programmation. Les structures de données exploitées sont des ensembles réalisés à partir de listes. Pour la deuxième année consécutive, le sujet n'évalue principalement que le programme de première année en algorithmique et programmation en ne permettant pas aux candidats de valider leur acquis dans le cadre des structures d'arbre.

Cette seconde partie est nettement plus longue que la première. Les questions posées sont de difficulté croissante. Cette partie a donc joué un rôle important pour distinguer les candidats.

Le groupement des parties logique et algorithmique a globalement permis de réduire la taille du sujet par rapport aux années 2002 et précédentes. Le sujet est toujours trop long. Par contre tous les candidats réussissent à aborder tous les aspects du sujet. L'ensemble des correcteurs a noté que : alors que les candidats disposent de trois heures pour traiter la totalité du sujet, ils ne traitent pas plus que ce qu'ils étaient capables de traiter en deux heures lors de l'introduction de l'épreuve d'option en Informatique. Ce point de vue est également partagé par les correcteurs de l'épreuve d'option en Sciences Industrielles.

Avant de commenter en détail chaque partie, certaines remarques peuvent être faites par rapport au comportement global de l'ensemble des candidats et son évolution par rapport aux années précédentes :

- l'impression que le niveau des copies semble avoir atteint un palier se confirme. Nous noterons toutefois un léger progrès en programmation par rapport aux années précédentes et une légère régression dans la qualité du traitement formel. Les questions simples sont en général bien traitées avec la rigueur nécessaire. Cette rigueur disparaît par contre trop souvent dès l'apparition de questions plus complexes. L'utilisation rigoureuse de l'induction

est encore absente d'un trop grand nombre de copies. Celle-ci est pourtant essentielle au traitement formel de l'Informatique. Nous insisterons sur le fait que les difficultés rencontrées par les candidats dans la rédaction des solutions sont maintenant bien identifiées que ce soit sur la forme ou sur le fond. Ceci devrait leur permettre de beaucoup mieux se préparer aux épreuves proposées dans le cadre du concours commun polytechnique. Étant donné que ce n'est toujours pas le cas, il semble maintenant nécessaire d'étudier finement les causes de la persistance de leur manque de préparation et d'en tirer les conséquences vis-à-vis du contenu et du volume des programmes ;

- l'impression que le rapport entre les copies exploitant CaML et celles exploitant PASCAL pour la programmation semble lui aussi se stabiliser. Nous noterons un progrès sensible dans les copies exploitant PASCAL. Malgré cela, la programmation en CaML étant plus efficace pour le type de problèmes traités dans le cadre du programme, la qualité moyenne des copies PASCAL reste inférieure à celle des copies CaML. Nous insisterons une fois de plus sur le fait que le langage PASCAL ne devrait plus être utilisé car cela va à l'encontre de l'intérêt des candidats. Il semble maintenant urgent de modifier le programme pour imposer définitivement CaML car même l'intérêt des candidats n'a pas conduit à son abandon dans les classes préparatoires ;
- nous insisterons une fois de plus sur l'extrême médiocrité de la rédaction des copies. Les candidats ne semblent toujours pas avoir compris qu'une réponse correcte mal rédigée sera en général considérée comme partiellement, voire totalement fautive. La qualité de la rédaction de documents scientifiques et techniques est pourtant un point essentiel du métier d'ingénieur auquel se destinent les candidats. En Informatique, les normes de qualité ISO 9001 concernent principalement la quantité et la qualité de la documentation. Les remarques issues du monde industriel concernant les ingénieurs débutants sont de plus en plus négatives sur ce sujet et il est important que les candidats en soient informés dans les classes préparatoires. L'objet n'est pas d'atteindre la perfection à cette période de leur formation mais de stopper la régression constatée depuis de nombreuses années.

Remarques particulières

Partie I - Automates et langages

La construction des automates et des expressions régulières décrivant le langage reconnu par un automate ne pose pas de problèmes. Les formalismes correspondant sont acquis par les candidats. Les seules erreurs fréquentes concernent les états initiaux ou terminaux dans la construction des automates ainsi que l'élimination des parties inaccessibles. Par contre, les preuves par induction sur les mots (récurrence sur la longueur du mot) sont encore trop souvent négligées au profit de raisonnements informels ou de manipulation de «...» pour représenter un mot de taille quelconque et une transformation appliquée aussi souvent que nécessaire. Cet aspect est maintenant inadmissible. L'exercice posé est classique et semblable à ceux qui ont été posés les années précédentes. Même le candidat qui n'a rien compris à l'induction devrait être capable de le traiter par mimétisme. Cette forme de preuve est essentielle en Informatique (ainsi qu'en Mathématique) et aucun candidat ne devrait atteindre ce niveau d'étude sans la maîtriser.

Question I.2 (construction des automates)

Les réponses fournies sont correctes dans la plupart des cas. Nous noterons que certains candidats ne respectent pas les indications du sujet et présentent des états et/ou des transitions inaccessibles. Les erreurs les plus fréquentes concernent les états initiaux et terminaux.

Certains candidats n'ont pas compris la signification des opérateurs de dérivation et proposent donc des réponses très éloignées des résultats attendus.

Question I.1, I.3 (description des langages)

Le mot vide est oublié par une part non négligeable des candidats. Les candidats qui ont conservé les états et transitions inutiles ont tendances à proposer des langages faux qui tentent de prendre en compte ces états et transitions. Certains correcteurs ont noté une confusion occasionnelle entre les opérateurs de répétition + et *.

Question I.4

Question I.5 (preuve par induction)

La notation proposée a perturbé certains candidats.

Question I.6

Certains candidats ne répondent pas tout en ayant traité correctement les questions précédentes. Il semblerait que ceux-ci n'aient pas du tout compris l'objectif de l'exercice pourtant fort classique dans les épreuves du concours commun polytechnique.

Partie II - Algorithmique et programmation

Le sujet proposait d'étudier une méthode d'interrogation de base de connaissance basée sur une procédure de complétion. La base de connaissance est représentée par une formule logique en forme normale. La procédure de complétion transforme cette forme normale pour simplifier l'interrogation.

Les correcteurs ont tous signalé que les candidats semblaient beaucoup plus à l'aise avec la programmation qu'avec le traitement rigoureux des algorithmes. Ce défaut usuel en Informatique est incompatible avec les systèmes complexes conçus actuellement. Il est essentiel de le corriger le plus tôt possible en insistant sur ces aspects tel que le fait le programme.

Les correcteurs ont majoritairement noté que la plupart des copies s'arrêtaient au niveau de la question II.20.

Les candidats utilisant CaML ont tendance à ne pas exploiter les fonctions fournies en intégrant celles-ci au niveau de chaque question.

Considérons séparément les deux thèmes du programme traités dans cette partie : la logique et la programmation.

Logique

La manipulation formelle des formules du calcul des propositions et/ou la construction de tables de vérité semblent acquises par la majorité des candidats. Nous noterons que le sujet ne contenait pas de question concernant la modélisation d'énoncés en langage naturel par des formules du calcul des propositions. Il est regrettable que cette part importante du programme n'ait pas été évaluée.

Question II.2, II.14, II.18, II.21, II.27, II.28, II.31 (exemples)

Les réponses sont en général correctes quand les questions ont été traitées. Nous noterons que les candidats ont en général donné des résultats bruts. Les correcteurs ont regretté de ne pas disposer dans certains cas de la manipulation explicite des formules pour pouvoir mieux comprendre certaines erreurs. La difficulté se pose toujours de savoir distinguer le nécessaire et le superflu. Certains candidats détaillent des calculs inutiles, d'autres sautent des étapes essentielles à la compréhension.

Question II.1, II.10, II.11, II.13, II.15, II.16, II.19, II.22, II.25, II.32, II.33 (preuve logique), II.26 (preuve algorithmique)

Les correcteurs ont noté qu'une très large majorité des candidats ne montrent pas la rigueur nécessaire à la rédaction d'une preuve et se contentent de reprendre le sujet en utilisant le résultat attendu comme hypothèse pour conclure par une pirouette. Le faible nombre de réponses satisfaisantes rendent ces questions très discriminantes.

Les correcteurs veulent insister une fois de plus sur l'importance des aspects formels de l'Informatique non seulement dans le cadre du programme mais surtout dans le métier d'ingénieur en Informatique. La conception et la réalisation de logiciels critiques dans un cadre industriel repose de plus en plus sur cette approche. Il est donc de plus en plus nécessaire que les candidats sachent construire des preuves correctes sans cacher la majorité des étapes et plus particulièrement les aspects non triviaux de la preuve.

Les correcteurs ont noté de nombreux raisonnements abusifs sous la forme de conditions nécessaires et suffisantes au lieu d'un traitement distinct avec des approches différentes pour les deux étapes.

Un nombre important de candidats simplifient de manière abusive «T» et «T renversé» et font ainsi disparaître les cas où les ensembles d'hypothèses ou de conclusions sont vides (II.10 et II.13).

Dans la question II.15, de nombreux candidats ne traitent que la condition évidente et oublient la réciproque.

De nombreux candidats ont inversé la relation $<$ définie en II.17 et exploitée dans les questions II.18 et II.19.

Programmation

Cette partie débute par la réalisation d'une structure d'ensemble qui est utilisée ensuite pour représenter la base de connaissance.

Les questions étaient de difficultés croissantes. Les questions demandant un traitement formel reposaient en général sur des exemples traités dans les questions précédentes.

Remarques générales :

Questions II.3, II.5, II.6, II.8, II.9, II.12, II.17, II.20, II.24, II.30, II.34 (programmation)

Ces questions sont de loin celles qui ont été traitées par le plus grand nombre de candidats. Les fonctions proposées sont généralement correctes.

Nous noterons que les candidats respectent bien les contraintes présentées en début de sujet interdisant la programmation impérative y compris les candidats formés à l'utilisation du langage PASCAL ce qui n'était pas systématiquement le cas les années précédentes.

Questions II.23, II.29 (explication des programmes)

Ces questions ont été très peu traitées probablement à cause de leur position dans le dernier tiers du sujet. Les candidats qui les ont traitées proposent des explications satisfaisantes en suivant le format requis. Ce progrès par rapport aux années précédentes semble dû au fait que, par manque de temps, seuls les candidats sachant répondre correctement ont choisi de traiter ces questions.

Questions II.4, II.7 (complexité)

Les questions posées étaient extrêmement simples. Elles ont été traitées de façon satisfaisante par les candidats qui les ont abordées.

Conclusion

L'épreuve organisée en deux parties distinctes a permis d'évaluer les capacités des candidats sur la majeure partie du programme. Le contenu des copies confirme le palier atteint les années précédentes dans la qualité des réponses. La difficulté relativement faible de la plupart des questions devrait permettre à tous les candidats de répondre de manière satisfaisante, au moins aux premières questions. Seule la vitesse ou la difficulté des dernières questions devraient jouer un rôle discriminant ce qui n'est malheureusement pas le cas. La programmation est globalement satisfaisante. Par contre, le traitement formel devrait encore progresser.

Les correcteurs n'ayant encore jamais participé à ce type d'épreuves ont été choqués par le comportement de la majorité des candidats. Leur impression est que « les candidats ne semblent pas avoir intégré le fait qu'une épreuve d'informatique est avant tout une épreuve scientifique quasi-mathématique : il est attendu d'eux une rigueur au moins équivalente à celle des épreuves de mathématique ».

La moyenne est de 6,32 et l'écart type de 2,67. Nous notons une baisse significative par rapport à l'année dernière ainsi qu'aux années précédentes. La question de la pertinence de la volonté de traiter une large part du programme conjointement avec un problème d'algorithmique pertinent se pose de manière de plus en plus criante. Il semble de plus en plus nécessaire de passer à une suite d'exercices indépendants dont le seul objectif serait la qualité de l'évaluation et pas un compromis entre celle-ci et l'intérêt et/ou le réalisme du sujet traité.

Il faut tout de même noter que malgré l'augmentation de la durée de l'épreuve et la réduction du nombre de parties distinctes, le nombre de questions traitées dans l'épreuve ne change pas de façon significative. Cette remarque semble augurer d'une baisse dans la performance des candidats vis-à-vis d'une épreuve de concours.

Nous concluons en remarquant que les recommandations évoquées dans les rapports depuis les premières sessions des épreuves optionnelles n'ont toujours pas été suivies d'effets significatifs notables dans les copies des candidats.