



CONCOURS PASS' Ingénieur

RAPPORT DE L'ÉPREUVE D'INFORMATIQUE

Les oraux d'informatique de la session 2024 du concours Pass' Ingénieur se sont tenus en deux sessions, les 25 et 26 mai sur le campus de l'ENSIC à Nancy et les 1^e et 2 juin sur le campus de l'ENSIACET à Toulouse.

17 candidats se sont présentés aux épreuves orales et ont été auditionnés au cours de cette session. Les notes à l'épreuve d'informatique sont allées de 7 à 18, avec une moyenne de 12,76 et un écart type de 3,51.

Soulignons la volonté des candidats de bien faire, de mettre en valeur leurs connaissances et les bonnes prestations d'ensemble, en moyenne, sur cette session. Cela a conduit à de bonnes prestations de candidats bien préparés, tandis que d'autres ont révélé de vraies lacunes dans leur maîtrise du cours, ou des difficultés importantes pour résoudre les problèmes posés.

Nous reprendrons dans ce rapport quelques rappels sur le déroulement de l'épreuve. Nous aborderons ensuite des conseils sur différentes parties du programme : questions théoriques, programmation, informatique.

Quelle que soit la formation universitaire du candidat, celui-ci peut être interrogé sur l'ensemble du programme de l'épreuve. Les candidats sont donc invités à lire attentivement le programme de l'épreuve d'informatique pour identifier les éventuels points non vus dans leur formation et s'y préparer.

Il arrive néanmoins que des candidats restent bloqués sur certains points de cours du programme lors de la préparation en autonomie, ou font des contre-sens (comme cette année, préparer l'algorithme de tri rapide, lorsque l'algorithme de tri fusion était demandé). Lors de la présentation, l'examineur peut alors, suivant le temps restant : **sauter les questions problématiques, remplacer ces questions par d'autres, donner des indications et des rappels de cours pour permettre la poursuite de l'épreuve.**

DÉROULÉ DE L'ÉPREUVE

Le candidat se voit remettre un sujet constitué de deux exercices indépendants (le second étant généralement plus long), portant sur des thèmes différents. Un exercice au moins comporte des questions de programmation.

Le candidat dispose de :

- 25 minutes de préparation pour tenter de résoudre ces exercices et préparer sa présentation,
- 25 minutes d'oral au tableau pour présenter ses résultats et poursuivre la résolution, avec les questions et indications de l'examineur.

L'épreuve ne comprend pas de partie sur machine : les programmes attendus sont écrits au tableau.

Enfin, les sujets proposés sont longs. La gestion du temps de préparation est donc importante. Il est conseillé au candidat de :

- ne pas rester bloqué par les questions non résolues durant la préparation,
- ne pas passer trop de temps à soigner la rédaction d'une réponse sur ses brouillons, que le candidat pourra développer à l'oral,
- veiller à aborder au moins une partie de chaque exercice, pour être ensuite capable de rebondir sur les indications de l'examineur à l'oral.

Le candidat est invité, durant le passage, à conduire lui-même l'oral : aborder les questions du sujet dans l'ordre qui lui convient, présenter ses résultats sans attendre de demandes ou questions orales de l'examineur.

PRÉCISIONS SUR LES QUESTIONS D'INFORMATIQUE THÉORIQUE

Nous avons remarqué beaucoup d'écarts dans la maîtrise des questions d'informatique théorique du programme. Il est à noter que certains points précis peuvent très bien être préparés en autonomie par des candidats n'ayant pas abordé certaines notions lors de leur cursus universitaire.

Ainsi, savoir dresser la table de vérité d'une formule booléenne, déterminer un automate fini, maîtriser la représentation des entiers et flottants en machine sont des notions importantes et leur méconnaissance peut bloquer les candidats sur certains exercices.

Lors de cette session, nous avons constaté une assez bonne maîtrise des automates finis et de leur détermination, même si le lien avec l'algorithmique du texte peut rester assez obscure pour certains candidats.

En logique booléenne, les notions de tautologie, satisfiabilité ne sont pas toujours maîtrisées et certains candidats n'ont pas de méthode claire pour étudier une formule, voire pour la noter.

La représentation en binaire est bien maîtrisée, mais la notion de flottant et de précision relative à l'ordre de grandeur des nombres n'est souvent pas maîtrisée. Ainsi, aucun candidat n'a su répondre seul à la question du comportement du programme suivant, la réponse généralement entendue étant que le programme bouclera à l'infini :

$a=1.0$

tant que ($a \neq a+1$) :

$a=a*2$

Les candidats ne sont pas attendus principalement sur le résultat, mais sur la méthode de résolution et sa présentation. Les candidats ont ainsi parfois tout intérêt, durant leur temps de préparation, à démarrer la résolution, vérifier leur méthode et passer à la suite sans achever la résolution, qu'ils pourront faire durant l'oral si l'examineur leur demande de poursuivre.

PRÉCISIONS SUR LES QUESTIONS DE PROGRAMMATION

Sur les deux exercices du sujet, l'un au moins comprend des questions de programmation pour lesquelles un code est attendu dans un langage donné.

Le candidat est invité, en début d'épreuve, à choisir librement son langage de programmation. Sur ce choix, il est conseillé de choisir le langage sur lequel le candidat est le plus à l'aise, sans calcul : l'examineur a le souci d'adapter les questions au langage et le choix d'un langage donné n'est pas pénalisant. Cette année, les candidats ont opté, dans l'ordre d'importance, pour le python, le C et le java, sans qu'il soit remarqué une meilleure maîtrise des candidats ayant opté pour tel ou tel langage.

Nous soulignons en général une bonne maîtrise de la syntaxe des langages choisis et une aisance à écrire du code correct de la plupart des candidats et ce dans tous les langages.

PRÉCISIONS SUR LES QUESTIONS D'ALGORITHMIQUE

Les candidats sont régulièrement confrontés, dans les sujets, à des problèmes pour lesquels des algorithmiques classiques sont utilisables : algorithmes gloutons, diviser pour régner, programmation dynamique, retour sur trace...

Nous précisons que ces techniques sont au programme ; un candidat ayant assimilé ces techniques sera naturellement bien mieux à même de suivre la résolution des problèmes proposés.

Lorsque le candidat ne sait pas proposer directement, seul, le bon algorithme, il est possible d'aborder le problème de diverses façons, qui permettent de ne pas rester bloqué et d'avancer dans la résolution :

- aborder l'algorithme sur des exemples, pour comprendre son déroulé,
- déterminer les données (variables) nécessaires,
- déterminer quel type de boucle (tant que, pour) il faudra utiliser,
- éventuellement, proposer une définition récursive du problème...

Les indications du sujet sont évidemment à prendre en considération : souvent, il ne s'agira pas de proposer un algorithme sans indications, mais d'aborder sa conception pas à pas, par exemple en suivant le déroulé sur un cas particulier, ou en proposant une relation de récurrence... guidant le candidat dans ces étapes.

Enfin, les questions d'analyse de programmes, telles que les preuves de terminaison ou l'analyse de complexité, ne sont pas à négliger. Sur ces questions aussi, le candidat n'est pas attendu principalement sur le résultat, comme l'ordre de grandeur de la complexité du programme, mais sur la méthode permettant d'estimer cet ordre de grandeur.