

Les oraux d'informatique de la session 2021 du concours Pass'Ingénieur se sont tenus en deux séries, les 29 et 30 mai 2021 à Chimie ParisTech et les 5 et 6 juin 2021 sur le campus de l'ENSIACET à Toulouse.

Le faible nombre de candidats dans cette épreuve d'informatique, spécifique de la filière Maths-Informatique, ne permet pas de dresser de généralités sur le profil des candidats. Mais nous pouvons souligner une certaine hétérogénéité des niveaux des candidats, au niveau de la maîtrise du programme comme de la capacité à résoudre des exercices et à prendre des initiatives.

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.

Nous rappelons que, quelle que soit la formation universitaire du candidat, celui-ci pourra 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.

## DÉROULÉ DE L'ÉPREUVE

Le candidat se voit remettre un sujet constitué de deux exercices indépendants, portant sur des thèmes différents.

Le candidat dispose de :

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

Notons que l'épreuve ne comprend pas de partie sur machine : les programmes attendus seront écrits au tableau.

Enfin, les sujets proposés sont volontairement longs. Il n'est absolument pas attendu que le candidat résolve toutes les questions, mais qu'il aborde au moins une partie de chaque exercice et soit capable de rebondir sur les indications de l'examinateur.

## PRÉCISIONS SUR LES QUESTIONS D'INFORMATIQUE THÉORIQUE

Les examinateurs constatent une maîtrise très inégale des questions d'informatique théorique au programme et souvent des soucis de vocabulaire rendant les questions inabordables même chez des candidats connaissant manifestement les notions.

Ainsi, une table de vérité d'une formule de la logique propositionnelle, ou la détermination d'un automate fini, sont des notions à connaître pour ne pas rester bloqué.

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.

## 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 – et à s'y tenir. Sur ce choix, nous conseillons 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 pour le C, le java et le python ; sans que les examinateurs ne remarquent une meilleure maîtrise des candidats ayant opté pour tel ou tel langage.

Nous observons que, chez certains candidats, la syntaxe du langage choisi n'est pas bien maîtrisée : que ce soit sur des mots-clés du langage, la façon de gérer les tableaux, etc. Si des coquilles sont bien sûr acceptables au tableau, une maîtrise de la syntaxe du langage est une compétence attendue et un prérequis pour aborder l'algorithmique correctement.

## 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... Les examinateurs constatent un blocage de nombre de candidats pour aborder de telles questions et un manque d'initiatives.
- Les indications du sujet sont évidemment à prendre en considération : souvent, il ne s'agira pas uniquement de proposer un algorithme, 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... Les candidats sont tous invités à se préparer à faire preuve d'initiative sur de telles questions : proposer soi-même un exemple test, par exemple, est bienvenu.
- Enfin, les questions d'analyse de programmes, telles que les preuves de terminaison ou l'analyse de complexité, ne sont pas à négliger. Proposer un invariant de boucle, ou estimer l'ordre de grandeur de la complexité d'une suite d'instruction, font partie des compétences tout à fait abordables, qui permettent de faire preuve de recul sur un programme donné.