



## CONCOURS PASS' Ingénieur

### RAPPORT DE L'ÉPREUVE D'INFORMATIQUE

Les oraux d'informatique de la session 2023 du concours Pass' Ingénieur se sont tenus en deux sessions, les 3 et 4 juin sur le campus de Chimie Paris Tech, et les 10 et 11 juin sur le campus de l'ENSIACET à Toulouse.

À noter un fort absentéisme, peu de candidats admissibles se sont présentés au cours de cette session. Les notes à l'épreuve d'informatique sont échelonnées de 4 à 17, avec une moyenne de 11,36 et un écart type de 4,22.

Nous notons une grande disparité de niveau de préparation des candidats, à la fois au niveau de leur maîtrise de l'ensemble du programme, que de leur capacité à prendre des initiatives, rebondir sur les indications des examinateurs et résoudre des exercices.

Nous soulignons que l'objectif de l'épreuve est, strictement, de déterminer au mieux les connaissances des candidats sur le programme de l'épreuve et leur capacité à mobiliser ces connaissances pour résoudre des problèmes informatiques, en utilisant au mieux le temps imparti.

Logiquement, les résultats reposent donc, avant tout, sur le travail durant les années d'étude en amont des épreuves ; néanmoins, l'attitude des candidats, leur capacité à se plier aux modalités de l'épreuve afin de mettre en valeur leurs connaissances et leur maîtrise des sujets durant le temps imparti aide les examinateurs à déterminer leur niveau et fait donc une différence.

Il arrive régulièrement qu'un candidat soit bloqué par une notion de cours, pour tout ou une partie d'un exercice, durant la préparation. L'examineur peut alors, suivant le temps et l'importance de la lacune, se concentrer sur le reste du sujet, ou donner des indications et éléments de cours pour débloquer le candidat et juger de sa capacité à rebondir dessus.

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 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.

## 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

Les examinateurs constatent une maîtrise très inégale des questions d'informatique théorique au programme ; nous soulignons que quelques notions et points de définition simples sont tout à fait préparables, en autonomie, pour les candidats n'ayant pas abordé ces notions dans leur cursus universitaire.

Ainsi, une table de vérité d'une formule de la logique propositionnelle, la représentation des nombres entiers et flottants en mémoire, ou la détermination d'un automate fini, sont des notions centrales 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 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 encore, 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 remarquons, pour cette session encore, une propension de nombre de candidats à négliger les bonnes pratiques généralement admises en programmation. Il convient de ne pas interrompre de boucle brutalement : on évitera ainsi les instructions *break* et *continue*, toujours dispensables.

## 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 soulignent 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.

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... Les candidats sont invités 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. 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.