

Éléments sur l'épreuve de TP à l'attention des candidat(e)s

4 mai 2022

Le présent document n'a pas de caractère réglementaire. Il se contente, en l'absence d'un rapport du jury avant la session 2022, de préciser les attentes du jury en vue de l'oral et de préciser la vision que le jury a de l'épreuve de TP, en complément du "sujet 0".

Pour alléger le texte, il n'a pas été adopté une formulation neutre ou l'usage de l'écriture inclusive ; on fait usage alternativement des formulations "la candidate" et "le candidat", mais cela doit être pris dans le sens du neutre : l'ensemble de ce texte, sans exception, concerne uniformément candidates et candidats.

Rappel de l'arrêté du 17 mai 2021 : *L'épreuve vise à s'assurer de la capacité du candidat à analyser et planifier son travail, à développer en conditions quasi-réelles, avec des outils standards, un code de qualité, à analyser un code produit par d'autres et à exposer et justifier les choix effectués en développant en particulier des problématiques de conception, de qualité du code et de garanties de comportement correct.*

L'épreuve consiste en la présentation devant le jury du travail effectué par le candidat, à partir d'un sujet fourni par le jury, suivie d'un entretien.

Le jury rappelle, au travers de cette épreuve, son attachement à une vision incarnée de l'informatique ne se limitant pas à de l'algorithmique ou de l'informatique théorique. La démarche de programmation est une partie inhérente de la pratique (et donc de l'enseignement) de l'informatique. La compréhension précise des aspects bas niveau et le recul requis pour pouvoir les enseigner nécessite de s'être confronté à leur mise en œuvre pratique : les sujets refléteront cette vision du jury.

Complément sur les sujets

- Les sujets sont construits de manière à permettre un minimum de progressivité ; si l'un des objectifs est d'observer la capacité des candidats à prendre des initiatives face à un problème relativement ouvert, le jury tient également à permettre à des candidats moins à l'aise d'avoir au moins une première partie à leur portée.

- À ce titre, les sujets s’efforcent de proposer une première partie d’ambition modeste, centrée autour d’un cœur algorithmique ; le sujet se développe ensuite autour de problématiques de plus bas niveau, en lien avec des aspects système, réseau, etc. L’objet de l’exercice n’est pas d’optimiser finement ce cœur algorithmique, et le jury préfère nettement voir les candidates démontrer une certaine maîtrise des problématiques “bas niveau”, ou tout au moins s’y confronter.

Complément sur les attentes du jury

- Le jury s’attend à un exposé de 30 minutes environ, dédié à la présentation des différentes parties du sujet (y compris entre 5 et 10 minutes pour la revue de code). Si le candidat estime ne pas pouvoir exposer suffisamment en détail son code, la méthode employée, les structures utilisées, la vérification, etc. durant le temps imparti, il est invité à suggérer quelques thèmes que le jury pourra demander de développer durant les 30 minutes de questions.
- La qualité du code produit est un élément central de l’évaluation. Un code de qualité est avant tout un code lisible et réutilisable, donc spécifié, structuré, modulaire, documenté et testé, sachant s’appuyer sur des bibliothèques existantes, incluant si possible des éléments garantissant sa correction. On renvoie en particulier à la partie 1, “Méthodes de programmation”, du programme des classes préparatoires MP2I-MPI. Le jury estime que divers styles de programmation ou méthodologies de développement peuvent permettre d’écrire un code de qualité.
- Une telle présentation est un exercice inhabituel, et doit à ce titre faire l’objet d’un travail de préparation spécifique en amont de l’épreuve. On veut à tout prix éviter une paraphrase linéaire du code et, à l’inverse, voir autant que possible des présentations montrant une certaine hauteur de vue sur la structuration générale du code. Ce sera l’occasion pour la candidate de mettre en avant ses connaissances, en expliquant par exemple les choix des structures de données et algorithmes mis en œuvre, le fonctionnement et les limitations des primitives bas niveau utilisées, etc.
- Concernant l’audit de code, le candidat sera confronté à des extraits de code qu’il devra comprendre, corriger au besoin, et pour lequel on attend des propositions en vue de le rendre plus lisible, plus facile à maintenir, voire plus efficace. Les trois langages au programme pourront être utilisés ici.

Complément sur l’organisation de l’épreuve

- Durant la présentation, on attend de la candidate qu’elle s’appuie sur la vidéoprojection du code produit et de démonstrations rapides du comportement du programme résultant. L’exposé pourra être complété de graphiques illustrant ce comportement. En plus d’un ordinateur et d’un vidéoprojecteur, la candidate aura un tableau à sa disposition.
- Une présentation basée sur des diapositives n’est pas demandée ni même souhaitée.