

**Barème.**

- Partie sur papier : chaque question sur 4 points, total sur 32 points, ramené sur 10 points, +10%.
- Partie sur machine : chaque question sur 2 points, total sur 22 points, ramené sur 10 points, +100%.

**Statistiques descriptives.**

Soit  $\varphi : \mathbb{R} \rightarrow \mathbb{R}$ ,  $x \mapsto \min\left(\frac{1}{10} \lceil 10x \rceil ; 20\right)$ .

	Papier	Machine	Note finale
Transformation	$p$	$m$	$\varphi\left(1, 1 \frac{10p}{32} + 2 \frac{10m}{22}\right)$
Note maximale	25	13,5	20+
Note minimale	2	0	3,2
Moyenne	$\approx 15,21$	$\approx 5,67$	$\approx 10,41$
Écart-type	$\approx 5,44$	$\approx 3,23$	$\approx 4,08$
Premier quartile	12	4	7,6
Médiane	16	5	9,9
Troisième quartile	18,5	7,5	12,7

**Remarques sur la partie papier.**

- Une fonction renvoie un résultat (retourne est un anglicisme).
  - On approche quelque chose (approximer est, au choix, un anglicisme ou un barbarisme).
  - On quitte les maths pour l'info, et les trois quarts des étudiants n'encadrent pas leurs résultats. Vu qu'il n'y aura que deux DS ce semestre, je n'ai pas mis zéro à ces copies, même si cela m'a démangé. Mais votre manque de maturité me déprime. Bien présenter votre copie vous fait gagner des points au concours. Si vous n'encadrez pas vos résultats, vous perdrez des points à chaque épreuve. C'est donc d'une stupidité sans nom.
- Q1** On vous demandait un indice du maximum, pas la valeur de ce maximum.  
Vous ne pouvez pas dire « l'indice du maximum ».
- Q2** Comme une récurrence, un invariant s'initialise (en début de boucle).  
Vous devez justifier ce qui s'apparente à une hérédité : montrez que si l'invariant est vrai à un tour de boucle, il est vrai au suivant.  
Ceux qui se sont trompés de 1 dans l'invariant (en écrivant par exemple : «  $T[\text{imax}]$  est le maximum de  $T[:i-1]$  ») auraient dû se rendre compte de leur erreur à l'initialisation de l'invariant.
- Q3** Simplifiez : on écrit  $O(n)$  plutôt que  $O(n-1)$ .  
Vous devez rédiger un minimum cette réponse.
- Q4** Un grand moment de solitude de mon côté.  
Pour la plupart, vous n'êtes pas arrivés à exprimer correctement le temps dans la relation de récurrence.
- Q6** J'attendais un résultat mathématisé. Ceux qui m'ont donné une version Python correcte n'ont pas été pénalisés.  
Cependant, mélanger maths et Python n'était pas du plus bel effet.
- Q7** Paraphraser le code écrit n'apporte rien. Pour justifier, écrivez des invariants de boucle.
- Q8** Là encore, vous devez rédiger un minimum cette réponse.

**Remarques sur la partie machine.**

Même si mettre en œuvre les méthodes numériques du cours peut être délicat, le traitement du fichier `pi.txt` ne l'était pas. Les questions 6, 7 et 8 n'étaient pas dures et rapportaient 6 points en tout. La moyenne de la partie machine est à 5,67 : c'est un peu inquiétant.

## Quelques extraits de rapports.

### Centrale 2016.

[...]

Au delà de la correction d'un algorithme ou d'une fonction, le jury attache une grande importance à la qualité du code : respect de l'indentation et de la syntaxe, utilisation de l'expressivité du langage pour obtenir des programmes concis, introduction de variables pertinentes, choix intelligent des noms de variables, décomposition d'un algorithme en plusieurs parties naturelles, etc. sont autant de savoir-faire qui rendent le code facile à lire et à comprendre. Le barème valorise ces qualités tout au long du problème.

[...]

Ces questions demandaient d'écrire quelques fonctions itératives simples, avec une ou plusieurs boucles, et ont été bien traitées par la majorité des candidats. Le sujet demandait de donner la complexité des fonctions demandées. Si la notion est bien comprise, son expression est trop souvent maladroite (« la complexité est  $9n^2 + 3n$  »), ou lourdement justifiée par un décompte exhaustif des opérations élémentaires qui s'étale parfois sur plusieurs pages, là où une analyse de la structure des boucles est tout aussi convaincante et efficace. De nombreux candidats oublient qu'une fonction de complexité linéaire dans une boucle conduit en principe à une complexité quadratique.

[...]

Pour illustrer ce propos, la question **III.B.4)** demandait de travailler sur les coordonnées et la vitesse d'un avion. 40% des candidats qui ont abordé la question ont posé :

```
id, x, y, z, vx, vy, vz, t0 = intrus
```

Il va de soi que la suite du code est claire et concise, et qu'elle évite des expressions telles que :

```
if intrus[1]*intrus[4]+intrus[2]*intrus[5]+intrus[3]*intrus[6]>0:
```

plus difficile à lire, comprendre, vérifier et même écrire. Le barème bonifie les réponses des candidats ayant pris cette initiative.

Autre illustration : la question **III.C.2)** demandait de trier une liste où une seule valeur n'était pas à sa place. Certains proposent une solution quadratique, ce qui est visiblement maladroit. D'autres une solution linéaire, ce qui est mieux. D'autres enfin s'attachent, dans l'esprit de la question, à minimiser le nombre d'opérations élémentaires. Il va de soi que le jury pondère ces différentes approches, même si elles répondent toutes à la question.

### Mines 2016.

[...]

Nous souhaitons rappeler aux candidats quelques conditions nécessaires pour espérer réussir cette épreuve :

- La maîtrise de la syntaxe de base des langages python et SQL est absolument indispensable. Le respect de l'indentation est capital, et la lisibilité de l'écriture est appréciée : il est rappelé que le doute ne bénéficie pas souvent au candidat.
- Les candidats sont invités à réfléchir sur les spécificités de chaque type informatique : une liste (ou une liste de listes) n'a pas les mêmes propriétés qu'un tableau numpy, par exemple.
- Les questions qualitatives, où un court paragraphe est attendu (par exemple la question 12) doivent être traitées avec soin : la concision nécessaire n'excuse pas l'imprécision parfois très gênante des termes utilisés, ou du raisonnement.

[...]

**Q2** : La notion d'invariant de boucle a été souvent maltraitée. Un certain nombre de candidats montre son ignorance presque complète de cette notion par des raisonnements tels que « si la liste est triée au début, alors elle est triée à la fin et donc c'est un invariant ». Parmi les candidats qui ont compris l'intérêt du raisonnement par récurrence, certains oublient l'importance de l'initialisation, ou bien d'utiliser l'invariant pour prouver que la liste est effectivement triée à la fin de l'exécution de la fonction.