

Devoir surveillé n° 1 - Remarques

Barème.

Toutes les questions sont notées sur 4 points, le total est de 112 points.

Statistiques descriptives.

	Note brute	Note finale
Note maximale	80	19
Note minimale	15	5
Moyenne	$\approx 39,31$	$\approx 10,22$
Écart-type	$\approx 16,05$	$\approx 3,47$

Optimisation de rendement d'une entreprise de livraison (CCINP 2021 TSI).

Il y avait beaucoup de questions simples, vous devez apprendre à les repérer, et bien les traiter. Il y a beaucoup de points à ramasser.

Les parties I.3 et I.4 ont été très peu abordées. Elles n'étaient pourtant pas plus compliquées.

1. Que de fautes d'orthographe dans certaines copies, c'est hallucinant ! On écrit « poids » même au singulier, une valeur est « maximale » ou « supérieure » avec un « e » etc ... Quelques uns d'entre vous ne connaissent ni le féminin ni le pluriel. Attention, certains concours notent le soin et l'orthographe.
3. L'énoncé pouvait sembler ambigu mais naturellement ici on attendait une cargaison respectant la contrainte de poids.
4. Dans cette question et beaucoup d'autres, j'ai lu énormément d'erreurs sur **range**. Je vous rappelle que par défaut **range** commence à zéro si vous ne donnez qu'un argument, et que la borne de fin n'est pas atteinte. Les listes commencent de plus à l'indice 0. Donc **L[len(L)]** n'existe pas. Faites très attention à cela. Vous pouvez trouver cela casse-pieds ou difficile, mais c'est la base en python ! Vous ne devez pas vous tromper sur les bornes, et ne pas oublier les éventuels ± 1 .
5. On ne voulait pas des ratios $\frac{v_i}{p_j}$ pour $i \neq j$.
7. Très mal traité. Les commandes sur les listes sont parfaitement transposables aux chaînes de caractères : **s[i]**, **len(s)**, par exemple. Deux chaînes de caractères peuvent également être comparées : **s1 < s2** teste si **s1** se trouve avant **s2** pour l'ordre lexicographique. Le seul problème est qu'on ne peut pas changer la valeur des caractères d'une chaîne : **str** n'est pas un type *mutable*.
8. Déterminer le type de ce tri est une question de cours de sup, basique. C'est un échec total. Trois élèves seulement ont su répondre, ce n'est pas normal. Vous devez savoir reconnaître les tris par insertion, sélection, fusion, tri rapide et par comptage. Par contre les complexités dans les pire et meilleur cas sont bien connues.
9. On ne demandait pas de modifier L : il fallait renvoyer une nouvelle liste. Là aussi, que d'erreurs sur les bornes de **range** et les indices.
21. La notion de clé primaire d'une table est très mal comprise. Il s'agit d'une manière de repérer sans ambiguïté et de manière unique une entrée de la table. La date ne pouvait pas convenir : on peut

très bien effectuer plusieurs livraisons à la même date. Idem pour l'heure, le numéro de client ou le numéro de local. Il ne pouvait donc pas y avoir de clé simple. De manière raisonnable, on avait besoin de 3 attributs pour avoir cette notion d'unicité : date, heure et numéro de client. Il s'agit donc d'une clé primaire *composée*.

- 22. On écrit `table.attribut` et non `attribut.table`. Donc ici `livraison.id_client`. Mais quand deux colonnes n'ont pas le même nom, il ne sert à rien de mettre le nom de la table devant. Par exemple `id_client = id` est très clair, pas besoin de `livraison.id_client = client.id`.
- 24. L'énoncé même de la question a été très mal compris, donnant lieu à des requêtes qui ne répondaient pas du tout à la question.
- 27. Ce qui était coûteux dans la question précédente et que l'on voulait améliorer, c'était le calcul des puissances de 2, et rien d'autre. Demander 2^{**1} , 2^{**2} ... 2^{**n} c'est effectuer $1+2+\dots+n = O(n^2)$ multiplications : c'est beaucoup ! Alors qu'on peut calculer 2^{**1} , le garder en mémoire puis avec une seule multiplication supplémentaire calculer 2^{**2} et ainsi de suite. De cette manière, calculer toutes ces puissances de 2 ne demande que n multiplications : c'est beaucoup mieux ! C'est un grand classique en informatique.