



CONCOURS COMMUN INP RAPPORT DE L'ÉPREUVE ÉCRITE 2023 D'INFORMATIQUE

1/ REMARQUES GÉNÉRALES

L'épreuve d'informatique du concours CCINP proposait de travailler sur la reconnaissance de caractères. La première partie traitait de l'acquisition du document en format numérique. La deuxième partie traitait de la rotation automatique du document, détection automatique des lignes, puis restauration des caractères bruités avant leur détection. La troisième partie traitait de la détermination des caractères par une méthode d'intelligence artificielle.

Le sujet permettait de balayer un large ensemble des compétences décrites dans le nouveau programme d'informatique en CPGE.

Cette session a poursuivi la mise en place d'un document réponse. Les candidats l'ont globalement bien utilisé. Il est rappelé qu'il est utile de réfléchir sur un brouillon afin de porter le plus proprement possible la solution sur le document.

La propreté des copies était globalement meilleure que les années précédentes. La qualité de la présentation et de la rédaction est prise en compte dans la notation.

Pour la lisibilité des programmes, il est vivement conseillé de choisir des noms de variables intelligibles, de bien soigner son indentation avec l'alignement sur les carreaux ou un trait vertical.

L'utilisation des commentaires, bien qu'apprécié, ne doit pas surcharger la copie et faire perdre trop de temps aux candidats. Certains bons candidats ont énormément commenté leur code mais cela a eu pour conséquence de ne pas aller très loin dans le sujet.

La gestion des parenthèses est souvent mal maîtrisée : oubli pour les fonctions (par exemple `range n :`), oubli dans les expressions mathématiques ($d^{**1/2}$ est différent de $d^{*(1/2)}$). On trouve encore des syntaxes `L.append(a,b)` pour ajouter deux éléments à une liste...

Même si les examinateurs sont parfois « souples » sur l'indentation, certaines erreurs d'indentation conduisent à des algorithmes faux.

2/ ANALYSE DÉTAILLÉE DES QUESTIONS

Q1 : Cette question a été assez mal traitée. Le calcul demandé pouvait être fait en ordre de grandeur, mais beaucoup d'erreurs.

Q2 : Des erreurs de typographie sont malheureusement restées dans le programme donné en exemple. Il fallait remplacer p par n0 et q par n1... D'autre part, la syntaxe `img[i,j]` est à remplacer par `img[i][j]`. Les examinateurs ont évidemment été souples sur les réponses faites par les candidats en acceptant les différentes interprétations possibles. Cependant, beaucoup de candidats se contentent de dire "double boucle" donc $O(n^2)$! Mais qui est n ? Une phrase pour dire que les instructions dans la deuxième boucle était en temps constant était également bienvenue.

Q3 : Cette question est assez bien traitée.

Q4 : Cette question est bien traitée.

Q5 : Cette question a été moyennement bien traitée. Les candidats ont eu un peu de mal à exploiter la fonction partiellement donnée.

Q6 : Cette question a été assez mal traitée. Il était attendu que se restreindre à 8 bits non signés permettait d'avoir un gain en terme de stockage par rapport à une représentation en nombre flottant par exemple. Et surtout la deuxième partie de la question, un nombre négatif ne peut pas être représenté avec un nombre non signé, le calcul donnera une représentation associée à un nombre positif (pas la valeur absolue contrairement à ce que beaucoup pensent) rendant le niveau de gris associé incorrect (figure 5 du sujet).

Q7 : Cette question a été globalement bien traitée. Les erreurs d'indentation et la position de la mise à zéro du compteur étaient ici particulièrement critiques.

Q8 : Cette question a été assez mal traitée. Il est vrai que la gestion du drapeau (`deb = -1`) pour déterminer si on est ou non à l'intérieur d'un bloc correspond à une manière de penser à laquelle certains candidats n'ont pas forcément été réceptifs, mais chercher à comprendre et à s'adapter à des codes écrits par d'autres fait aussi partie des objectifs de la formation.

Q9 : Cette question a été souvent traitée. Si le nom de la dichotomie a été assez largement donné, la justification de l'arrêt de l'algorithme a parfois été farfelue : "l'algorithme s'arrête car il y a un while avec un epsilon" n'est pas une réponse ! Le nombre d'itérations quand il a été calculé a souvent été correct.

Q10 : Cette question a été assez bien traitée. Toutefois, le calcul du milieu des segments est assez souvent faux : $(b-a)/2$, $a*b$, ...

Q11 : Certains candidats font vraiment preuve de beaucoup d'imagination pour rendre le graphe illisible. Ne pas hésiter à utiliser un brouillon !

Q12 : Cette question est assez bien traitée par le peu de candidats ayant abordé la partie sur les graphes.

Q13 : Question plutôt mal traitée. Cette question nécessitait de bien comprendre l'algorithme expliqué. Une ambiguïté dans le choix du chemin était restée dans le sujet. Il fallait comprendre qu'on choisit le plus court chemin en nombre d'arêtes. En cas d'égalité en nombre d'arêtes, on choisit celui dont la somme des étiquettes est la plus faible. Toutefois, l'algorithme fonctionnait quand même en comprenant le plus court en somme d'étiquettes, mais nécessitait plus d'étapes. Les examinateurs ont noté juste toute solution qui suivait la logique comprise par le candidat...

Q14 : Il fallait calculer la coupe sur le résultat de la Q13.

Q15 : Peu de candidats ont compris que le but de l'algorithme était de séparer les pixels en deux groupes : les noirs et les blancs !

Q16/Q17/Q18 : Les questions de SQL ont donné un second souffle à beaucoup de candidats. Les questions sont assez bien traitées. On notera toutefois les erreurs suivantes :

- attribut.TABLE au lieu de TABLE.attribut ;
- BETWEEN mal maîtrisé auquel on pourra préférer une double comparaison ;
- dans les jointures, bien lier les bons identifiants ;
- beaucoup de WHERE après le GROUP BY.

Q19 : Cette question n'a pas été aussi bien réussie que ce qu'elle aurait dû être. Il convient d'être précis quant au type des valeurs retournées. Cela est certainement dû à de la précipitation, la fin de l'épreuve étant sans doute proche !

Q20 : On notera une maîtrise assez approximative des dictionnaires. Très très peu de candidats pensent à initialiser le dictionnaire si la clé n'est pas présente et font directement un append sur un objet inexistant.

Q21 : Question souvent bien traitée.

Q22 : Cette question a été assez peu traitée et souvent mal comprise.

Q23 : Les examinateurs attendent d'un candidat de connaître le nom d'un tri naïf et d'un tri performant avec la complexité associée.

Q24 : Question très mal traitée. Le sujet aurait certainement pu être plus clair sur la méthode en expliquant le tri par insertion.

Q25 : Question peu traitée et dépendante de la question précédente.

Q26 : Très peu de candidats se sont lancés dans cette question, mais certains l'ont fait correctement.

Q27 : Une très grande partie des candidats ont analysé les résultats obtenus et fourni une conclusion adéquate.