PSI

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

1/ REMARQUES GÉNÉRALES

L'épreuve d'informatique du concours CCINP proposait de travailler sur les montres connectées et l'analyse de quelques données qui leur sont propres. La première partie traitait du traitement et de l'analyse des données GPS et faisait appel essentiellement à des compétences sur le traitement des chaines de caractères. La deuxième partie traitait de l'acquisition du rythme cardiaque. La dernière partie traitait du partage des données.

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

L'épreuve d'informatique est une épreuve scientifique dans laquelle les candidats doivent conserver leurs bonnes habitudes scientifiques : les résultats numériques doivent être accompagnés de leur unité (avec les symboles du Système International et non par exemple sec pour s), le nombre de chiffres significatifs doit être raisonnable (il convient d'éviter les réponses du type "environ 20 234 ko" après avoir estimé un ordre de grandeur). Les résultats doivent être homogènes (par exemple à la Q12, G est une grandeur sans dimension et non une fréquence ou autre).

Cette session a poursuivi la mise en place d'un document réponse. Les candidats l'ont globalement bien utilisé.

Pour cette troisième session en correction numérique, bien que la majorité des copies étaient assez propres, les correcteurs remarquent qu'un nombre non négligeable était très sale. L'épreuve comporte un nombre de questions limité permettant au candidat de réfléchir au brouillon pour fournir une solution propre sur le document réponse. 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'épreuve a été correctement réussie par la majorité des candidats. Les correcteurs ont pu constater quelques erreurs nouvelles dans la syntaxe du langage Python. Notamment des tests avec des tuples, le test d'égalité (==) est souvent confondu avec l'affection (=).

Beaucoup de programmes, sortent de manière prématurée avec des return directement dans les boucles :

```
for k in range(n):
    if L[k]=='' :
        return True
    else :
        return False
```

Les correcteurs ont été plus intransigeants cette année au type de retour des fonctions : quand un booléen est demandé, les chaînes de caractères « True » « False » ne sont pas acceptées ; quand une liste est demandée « return a,b » est compté faux…

On retrouve toujours quelques erreurs ou maladresses : des décalages de + 1 ou - 1 sur les indices dans les boucles et les listes, des oublis de définition de certaines variables utilisées dans les fonctions. La confusion entre l'indice et l'élément dans une boucle : « for i in temps » puis un accès à « temps[i] ».

La gestion des parenthèses est souvent mal maîtrisée : oubli pour les fonctions (ie range n :) ; oubli dans les expressions mathématiques (1 / 1+G est différent de 1 / (1+G)).

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

Certains candidats ont une tendance à alourdir leur programme en recopiant systématiquement les variables locales ou en introduisant trop de variables intermédiaires.

2/ REMARQUES SPÉCIFIQUES

Q1. Cette question consistait à lire une fonction donnée et d'en expliquer les différentes étapes. Il fallait le temps de bien lire le sujet, les annexes pour bien comprendre les différentes étapes. Notons que la compréhension globale des données manipulées a été assez mal comprise ici et dans la suite à cause d'un temps d'appropriation certainement insuffisant. Cette question a été globalement bien traitée.

Q2. La majorité des candidats n'a pas compris que l'argument était une liste de chaines de caractères et que l'on cherchait l'indice dans cette liste de la chaine contenant '$GPPGA'. La recherche d'un motif avec l'opérateur « in » ne semble pas connue. Une boucle WHILE était explicitement attendue mais ignorée par certains candidats. De nombreuses réponses ont imbriquées des for avec des while (ou l'inverse)…

Q3. Pourtant très simple, l'affectation dans une variable du retour d'une fonction ne semble pas être maitrisée.

Q4. Les candidats recherchent assez rarement la chaine de caractère vide '' mais plutôt une liste vide [] ou None.

Q5. Les candidats pensent assez rarement à convertir dans un type supportant la comparaison numérique avant de l'effectuer.

Q6. Cette question d'un niveau de difficulté assez élevé a été très peu traitée. Toutes les informations nécessaires étaient données dans les annexes et la question.

Q7. Outre les oublis de conversion en flottant, l'extraction des sous-chaines correspondant aux heures, minutes et secondes est assez mal réalisée… Extraction caractère par caractère par exemple. Le slicing semble peu maîtrisé pour une partie des candidats.

Q8. Encore une fois, une bonne lecture de l'annexe permettait de bien comprendre comment était la chaine de caractères relative aux angles. Nous retrouvons les mêmes oublis de conversion, les soucis de passage en minutes d'angle également. Pratiquement aucun candidat n'a tenu compte de la remarque comme quoi les degrés étaient avec 2 **OU** 3 caractères, rendant la majorité des extractions fausses.

Q9. Cette question est plutôt bien traitée par les candidats quand ils pensent à bien retrouver une liste.

Q10. Très peu de candidats arrivent à compter le bon nombre de caractères donné dans l'exemple…

Q11. On retrouve assez souvent la solution de supprimer les données au fur et à mesure… Difficile d'avoir 200 h de sauvegarde si on efface !

Q12. Trop peu de candidats réussissent à exprimer l'intégrale par la méthode des trapèzes !

Q13. Question assez bien traitée quand elle a été abordée. La principale erreur étant de faire une boucle trop grande en len(e).

Q14. Très peu de candidats réussissent à donner le test complet ; un certain nombre met des AND entre chaque test au lieu des OR, une possibilité consistait à mettre simplement NOT (condition globale à trouver).

Q15. Cette question a globalement été bien traitée.

Q16. Cette question a globalement été bien traitée avec une double boucle assez bien gérée et la gestion du calcul. En revanche, l'initialisation de la sous-partie est souvent faite avant la 1re boucle et non à chaque passage rendant le calcul faux !

Q17. Assez peu de candidats pensent à évoquer la composante moyenne du signal. Les réponses sont souvent trop générales. Il y a beaucoup de confusion entre valeur moyenne et fondamental d'un signal.

Q18. Certains candidats trouvent les bons indices à prendre en compte, mais placent mal la modification. ils font le calcul complet de S(k), puis se posent la question de savoir s'ils l'ajoutent dans la liste finale !

Q19. Question plutôt mal traitée arrivant en fin d'épreuve, les candidats n'ont sans doute pas eu le temps de bien comprendre le programme.

Q20. Il semblerait que les réponses soient données au hasard la plupart du temps au vu du nombre de bonnes réponses obtenues.

Q21. Les correcteurs attendaient l'instruction à ajouter dans le programme pour utiliser la fenêtre de Hann et non une phrase ; surtout quand la phrase consistait à dire qu'il faut utiliser la fonction Hann sans rien préciser !

Q22. Cette question a été plutôt bien traitée.

Q23. Cette question a été plutôt bien traitée. Le changement d'unité pose parfois quelques soucis.

Q24. Cette requête SQL un peu difficile a donné du mal à beaucoup de candidats qui n'ont pas su bien expliquer ce qu'elle renvoyait.