

Questions spécifiques



Le but de cette fiche est de montrer quelles questions spécifiques peuvent apparaître dans les sujets de bac à l'épreuve écrite. Les questions sont déconnectées de leur contexte complet, mais celui-ci n'est en principe pas nécessaire pour y répondre.

Exercice 1 (Processus)

Citer les trois états dans lesquels un processus peut se trouver.

Exercice 2 (Données)

Voici un extrait du fichier `flashcards.csv` réalisé par l'étudiante :

```
discipline;chapitre;question;r ponse
histoire;crise de 1929;jeudi noir - date;24 octobre 1929
histoire;crise de 1929;jeudi noir - quoi;krach boursier
histoire;2GM;l'Axe;Allemagne, Italie, Japon
histoire;2GM;les Alli s;Chine,  tats-Unis, France, Royaume-Uni, URSS
histoire;2GM;Pearl Harbor - date;7 d cembre 1941
philosophie;travail;Marx;ali n nation de l'ouvrier
philosophie;travail;Beauvoir;donne de la valeur   l'homme
philosophie;travail;Locke;permet de fonder le droit de propri t 
philosophie;travail;Crawford;satisfaction et estime de soi
```

1. Donner le s parateur choisi par l tudiante pour son fichier `flashcards.csv`.
2. Justifier pourquoi l tudiante a choisi ce s parateur.

Exercice 3 (Bases de donn es – Assez fr quent)

Deux questions similaires sur le m me th me :

1. Citer deux avantages   utiliser une base de donn es relationnelle plut t qu'un fichier texte ou un fichier tableur.
2. Expliquer l'int r t d'utiliser deux tables (`Personnel` et `Centre`) au lieu de regrouper toutes les informations dans une seule table.
3. Expliquer l'int r t d'utiliser trois tables `Chanson`, `Album` et `Groupe` au lieu de regrouper toutes les informations dans une seule table.

Exercice 4 (Complexit )

Sans l' nonc  complet il est impossible de r pondre   cette question, cependant il s'agit d'un type de question qui appara t :

Expliquer si ce code est adapt    une telle quantit  de sites   traiter. Justifier votre r ponse.

Il s'agit en fait d' valuer le co t (la complexit  en temps) du code, et de dire s'il est raisonnable ou non.

Exercice 5 (M thode algorithmique)

Lors de l'appel de `nb_solutions(score)`, on se rend compte que le nombre d'appels r cursif augmente tr s rapidement lorsque `score` augmente. Nommer une m thode algorithmique optimisant le nombre d'appels r cursifs.

Exercice 6 (M thode algorithmique)

Expliquer en quoi consiste le fait de trouver une valeur par recherche exhaustive.

Exercice 7 (Codage)

On considère que chacune des quatre aptitudes aurait pu être codée par une chaîne de 10 caractères dont chaque caractère utilise 1 octet en mémoire.

Choisir, avec justification, l'économie mémoire que le codage sur un entier de 8 bits permet de faire comparée au codage basé sur les chaînes de caractères : environ 10%, 30%, 50%, 98%.

Exercice 8 (Chiffrement)

Expliquer brièvement ce qu'est un algorithme de chiffrement symétrique/asymétrique.

Exercice 9 (Algorithme de tri)

On considère la fonction de tri suivante (à compléter) :

```
def tri(tab: list) -> None:
    n = len(tab)
    for i in range(1, n):
        marchandise = tab[i]
        j = i-1
        while ... and ... > ... :
            tab[j+1] = ...
            j = ...
        tab[j+1] = marchandise
```

Sans justifier, préciser le nom de ce tri, ainsi que son coût temporel dans le pire des cas (constant, logarithmique, linéaire, quasi-linéaire ($n \log_2(n)$), quadratique, cubique ou exponentiel).

Exercice 10 (Graphes)

Expliquer en quoi le choix fait par le pisteur-secouriste de choisir la distance minimale pour arriver le plus rapidement possible sur le lieu de l'incident est discutable. Proposer un meilleur critère de choix.

Exercice 11 (Piles et files)

1. Indiquer la signification des lettres dans les acronymes LIFO et FIFO.
2. Indiquer l'acronyme utilisé pour désigner la structure de file.