Examen Client/Serveur, 1995/1996

Tout document autorisé

Une réponse correcte ne rapporte pas de points si vous n'argumentez pas. Pour la plupart des questions, répondez d'abord par des généralités, des exemples réels ou comment vous feriez. Si la question vous semble fausse ou absurde, alors *prouvez-le*, cela vous dispensera de répondre. Les questions ne sont pas ordonnées.

1. Dans le cas ou le client et le serveur fonctionnent sur la même machine. Qu'est-ce que cela apporte par rapport a un programme monolithique ?
2. Est-t-il facile de transformer un programme bien structuré en une partie client et une serveur ?
3. Est-t-il facile de transformer un programme bien structuré dans un langage objet pour le mettre dans une architecture client/serveur ?
4. Quel est le grand principe de base à appliquer pour que la communication entre le client et le serveur soit résistante aux pannes du serveur ?
5. Quels concepts/outils sont importants pour assurer la communication client/serveur entre machines complètement différentes ?
6. Dans le monde des systèmes d'exploitation, expliquez la différence entre une architecture client/serveur 2 tiers et 3 tiers.
7. Citez des applications informatiques ne pouvant pas être découpées suivant la logique client serveur.
8. Que se passe-t-il (et à quoi cela pourrait servir) si l'on fait :
   `telnet b710siaa 6000`
9. Quel outil de base faut-il utiliser si l'on veut avoir un flot de données bidirectionnel sans perte entre 2 machines TCPIP ?
10. Quel outil de base faut-il utiliser pour faire exécuter une action élémentaire (indépendante des précédentes et suivantes) par le serveur ?
11. Quel sont les 2 seules méthodes générales permettant l'authentification (et le cryptage) sur des réseaux non protégés ?
12. Dans le cas de liaisons lentes en débit, que faut-il prévoir dans le protocole de communication client/serveur pour avoir de bonnes performances ?
13. Dans le cas de liaisons lentes en temps de latence (liaison satelite par exemple), que faut-il prévoir dans le protocole de communication client/serveur pour avoir de bonnes performances ?
14. Dans le cas de serveur d'informations en lecture, comment faire pour que le service ne soit pas interrompu lorsque le serveur est en panne ou impossible à accéder ?
15. Dans le cas de serveur d'informations en lecture/écriture, comment faire pour que le service ne soit pas interrompu lorsque le serveur est en panne ou impossible à accéder ?
16. Donner des exemples de serveurs pouvant être clients d'eux-même. Pourquoi le faire ?
17. Pour des accès SQL en "lecture" à un serveur de base de données. Peut-on faire un *PROXY cache* ?
18. Que faut-il faire pour que votre architecture client/serveur puisse évoluer dans le temps ?
19. Peut-on faire une architecture client/serveur qui s'adapte automatiquement pour avoir des performances optimums en fonction de l'utilisation qui en est faite ?
20. Pourquoi de nombreux protocoles clients/serveurs sont en ASCII lisible par un être humain ?
21. ODBC a été conçu pour rapporter de l'argent a MICROSOFT et non pour être une solution "ouverte". Argumentez cette affirmation et proposez une solution ouverte.
22. Que doit on faire pour augmenter l'interactivité (diminuer le temps de réponse) dans une architecture client/serveur. Cette solution ne va-t-elle pas à l'encontre de la notion même de client/serveur?
23. Lorsqu'une connexion bas niveau a lieu entre un client et un serveur, que doivent-ils faire ensemble avant de réellement travailler?