

Licence 2 — Info4B

Examen du Jeudi 9 Mai 2019

Durée 2h • Documents autorisés : une feuille A4 recto-verso

Le barème est donné à titre indicatif



Si un exercice vous conduit à faire des hypothèses, indiquez-les clairement sur votre copie.

Rédigez et justifiez précisément les réponses aux questions.

Exercice 1 - 6 pts

Les réponses aux questions de cet exercice sont limitées à 1/4 page de texte par question (hors schémas et/ou extraits de code).

1. Les méthodes `wait()` et `notify()` sont associées au principe des moniteurs en Java. D'une manière générale, que permettent-elles ?
2. Définir la notion de *buffer* (tampon) et décrire son utilité.
3. Dans le cas où une ressource de type tampon est utilisée par plusieurs threads, quel est le modèle d'interaction correspondant (deux modèles peuvent convenir, en décrire au moins un).
4. Décrire trois méthodes de communication entre threads.
5. Le noyau du système d'exploitation gère les zones de mémoire allouées et libres. À un instant donné, il possède une liste de 3 zones mémoire libres (non consécutives) de 32, 20, 65 blocs. Chaque bloc mémoire a une taille de 1 024 octets. Un processus demande une allocation de 20 224 octets. Avec l'algorithme *first-fit* quelle zone sera choisie et quelles sont les tailles des fragments internes et externes produits. Même question avec l'algorithme *best-fit*.

Exercice 2 - 6 pts

On considère une mémoire de 16 pages (numérotées de 0 à 15) comportant 10 pages de mémoire virtuelle simulée par un disque dur. Chaque page a une taille de 512 octets.

1. Faire un schéma de l'organisation de la mémoire, préciser la limite de la mémoire physique, numéroter les pages et donner les adresses des limites de page.
2. On suppose que le mécanisme de *swap* utilise une stratégie LRU (*Least Recently Used*), c'est-à-dire moins récemment utilisé. Donner les différents états de la mémoire physique en fonction des 14 demandes d'accès mémoire du tableau 1. On suppose que dans l'état initial la mémoire physique contient, dans l'ordre, les pages 5, 8, 2, 4, 1, 3. Ceci correspond également à leur ordre d'utilisation.
3. Pour chaque accès, donner le couple numéro de page, offset et le couple correspondant à la page placée en mémoire physique.
4. Quel est le nombre de défauts de page ?
5. Quels composants gèrent les défauts de page et les translations d'adresses dus au swap ?

demande	1	2	3	4	5	6	7
accès adresse	330	2 500	3 010	4 500	6 210	1 600	430
demande	8	9	10	11	12	13	14
accès adresse	6 100	6 400	7 200	3 002	2 060	180	2 400

TABLE 1 – Demandes d'accès à la mémoire

Exercice 3 - 8 pts

On développe une application réseau pour permettre à des utilisateurs de poster des photographies. À partir d'un programme client, l'utilisateur envoie une photographie à un serveur sous la forme d'un objet de la classe `Photo` contenant son nom d'utilisateur, un tableau d'octets (l'image), le nom de l'image incluant son format (jpg, png, etc.) et une liste d'annotations (tags ou hashtags). Le serveur range ces objets dans une structure de données permettant de retrouver toutes les photographies postées par un utilisateur et celles correspondant à une annotation. Le serveur doit supporter plusieurs clients simultanés.

1. Faire un schéma et y faire figurer le serveur, les clients, les threads, les structures de données partagées, les fichiers, etc. Décrire et justifier les constructions Java que vous allez utiliser pour implanter votre système.
2. Quel type de socket utilisez-vous pour les connexions entre les clients et le serveur ?
3. Écrire le squelette de la classe `Photo` (attributs et nom des méthodes, paramètres, sans le corps des méthodes).
4. Écrire le code de la méthode de la classe `Photo` qui permet de lire un fichier contenant une image.
5. Écrire l'extrait de code qui permet au serveur de prendre en charge plusieurs clients et de réceptionner des objets de la classe `Photo`.
6. Définir la classe `BasePhoto` du programme serveur qui permet de gérer et d'organiser les objets de la classe `Photo` (ajouter, rechercher par nom d'utilisateur, par annotation). Vous ne donnerez que les attributs, les noms des méthodes, leurs paramètres et leurs modificateurs (le code des méthodes n'est pas demandé). Indications : vous devez utiliser des tables de hachage, la classe `BasePhoto` ne contient qu'un objet.
7. Expliquer les problèmes de concurrence éventuels côté serveur et comment les résoudre ?
8. Écrire l'extrait de code qui permet d'ajouter une photo dans la classe `BasePhoto`.
9. Comment stocker l'objet de la classe `BasePhoto` dans un fichier ?