

Programmation Système

Juliusz Chroboczek

11 Janvier 2007

La durée de l'examen est de 3 heures. Les parties sont indépendantes. Les documents sont autorisés, le matériel électronique est interdit.

Tous les programmes demandés devront être écrits en C. Les solutions dans d'autres langages de programmation ne seront probablement pas acceptées.

La notation prendra en compte la qualité de la gestion des situations exceptionnelles (« erreurs ») ou l'absence de celle-ci.

Il est fortement conseillé de lire l'énoncé, qui consiste de 3 pages.

1 Questions de cours

Quel est le contenu du fichier toto après l'exécution de chacun des fragments de code suivants? Vous n'avez pas à justifier votre réponse. (On supposera les variables fd et fd2 déclarées et tout le reste de la bureaucratie habituelle effectué.)

1.

```
fd = open("toto", O_CREAT | O_WRONLY | O_TRUNC, 0666);
write(fd, "abc", 3);
close(fd);
```

2.

```
fd = open("toto", O_CREAT | O_WRONLY | O_TRUNC, 0666);
write(fd, "abc", 3);
lseek(fd, 2, SEEK_SET);
write(fd, "d", 1);
close(fd);
```

3.

```
fd = open("toto", O_CREAT | O_WRONLY | O_TRUNC, 0666);
write(fd, "abc", 3);
lseek(fd, -2, SEEK_CUR);
write(fd, "d", 1);
close(fd);
```

4.

```
fd = open("toto", O_CREAT | O_WRONLY | O_TRUNC, 0666);
write(fd, "abc", 3);
lseek(fd, 2, SEEK_END);
write(fd, "d", 1);
close(fd);
```

5. (*Optionnel, hors barème*)

```
fd = open("toto", O_CREAT | O_WRONLY | O_TRUNC, 0666);
fd2 = dup(fd);
write(fd, "abc", 3);
write(fd2, "d", 1);
close(fd);
```

2 Entrées/sorties fichiers

Dans cet exercice, l'utilisation de la bibliothèque `stdio` n'est pas autorisée, sauf pour présenter des messages d'erreur à l'utilisateur.

La plupart des formats de fichiers binaires commencent par un *entête*; cet entête contient normalement à son début une *signature*, une suite d'octets qui permet d'identifier le type de fichier. Le format « .zip » d'archivage des fichiers ne contient pas d'entête, mais plutôt un *enqueue* qui se trouve à la fin du fichier.

L'enqueue d'un tel fichier a une longueur de 22 octets, et commence par les quatre octets (en hexadécimal) 50 4b 05 06. À 10 octets du début de l'enqueue (donc à 12 octets de la fin du fichier) se trouvent deux octets qui indiquent le nombre de fichiers contenus dans l'archive, en format petit-boutiste (*little-endian*). (*Indication : si vous n'avez rien compris à ce paragraphe, faites un dessin.*)

1. Écrivez un programme qui prend en paramètre le nom d'un fichier *f* et vérifie si *f* est une archive « .zip » en vérifiant la signature qui se trouve au début de l'enqueue.
2. Modifiez votre programme pour qu'il affiche aussi le nombre de fichiers contenus dans l'archive.

3 Parcours d'arbre

Dans cet exercice, l'utilisation des bibliothèques `ftw` et `fts` n'est pas autorisée.

1. Écrivez un programme qui parcourt l'arborescence enracinée au répertoire *home* de l'utilisateur courant et donne la liste des fichiers de taille supérieure à 1 Mo.

Indication : le nom du répertoire home de l'utilisateur courant est stocké dans la variable d'environnement « HOME ».

2. Dites *en quelques mots* ce que fera votre programme s'il rencontre un lien symbolique pointant sur un fichier de taille supérieure à 1 Mo.

3. (*Optionnel, hors barême.*) Modifiez votre programme pour que, lorsqu'il rencontre un lien symbolique pointant sur un fichier de taille supérieure à 1 Mo, il indique le nom du lien symbolique ainsi que le nom du fichier sur lequel il pointe. (*Attention, un lien symbolique peut contenir un chemin absolu ou un chemin relatif au répertoire contenant le lien.*)

4 Processus

Dans cet exercice, l'utilisation de la fonction `system` n'est pas autorisée.

La commande `whoami` affiche le nom de l'utilisateur qui l'exécute. Par exemple,

```
$ whoami
jch
$
```

Écrivez un programme C qui prend en paramètre un nom de fichier *f* et qui écrit le nom de l'utilisateur qui l'exécute dans le fichier *f*. Votre programme ne devra pas exécuter de programmes autres que « `whoami` ».

*Indication : on pourra par exemple exécuter « `whoami` » avec la sortie standard redirigée sur le fichier *f*.*