

TP de Système n° 5

On vous rappelle qu'à partir de maintenant, les réportoirs de chaque TP devront être accompagnés d'un Makfile permettant de compiler chaque exercice, et un cible "11" pour les compiler tous.

Exercice 1 : "stat"

Écrire la commande "stat(1)" à l'aide de l'appel système stat(2)

```
int stat(const char *path, struct stat *buf);
```

qui permet de stocker les informations relatives au fichier path dans la structure pointée par buf : buf est un pointeur sur une structure stat, qui contient entre autres le champ st_mode de type mode_t ; ce champ définit à la fois le type du fichier et les droits d'accès au fichier (voir la page de manuel). Elle prendra un ou plusieurs fichiers en argument et devra afficher pour chacun :

1. Son type (fichier, répertoire, lien symbolique ...)
2. Son numéro d'inode ,
3. Le nombre de liens durs pointant vers cet inode ,
4. Sa taille ,
5. Ses permissions, dans le style de "ls -l".

Exercice 2 : "fortune"

Le programme "fortune" permet d'afficher à l'écran une citation tirée aléatoirement dans un fichier *indexé*, c'est-à-dire composé d'un fichier principal **fortune.d** contenant les citations juxtaposées et d'un fichier **fortune.idx** contenant les positions des citations dans **fortune.d**.

1. Écrire un programme **makefortune** créant **fortune.d** et **fortune.idx** à partir d'un fichier **fortune.txt**, dans lequel chaque citation est séparée de la suivante par un ligne contenant uniquement le caractère %.
2. Écrire ensuite le programme **fortune** affichant une citation aléatoire.

Exercice 3 : "cp -p"

Implémenter la commande "cp -p source dest" qui copie le fichier *source* vers le répertoire ou nom de fichier *dest* en préservant les permissions, les propriétaires et group, les dates. Si *dest* est un répertoire, on créera le fichier *dest/source* ; sinon on copiera *source* dans *dest*.

1. Écrire une fonction qui teste si son paramètre est le nom d'un fichier ordinaire, d'un répertoire ou d'un fichier d'un autre type.
2. Écrire un programme prenant deux paramètres *source* et *dest*, vérifiant que *source* est un fichier ordinaire, puis tentant de créer un copie en préservant les droits d'accès sous le nom *dest/source* ou *dest* selon que *dest* est un répertoire ou non ; cette copie ne sera pas créée si le fichier d'destination existe déjà.
3. Modifier votre programme pour qu'il gère l'option **-f** permettant l'écrasement d'un fichier ordinaire préexistant.
4. Modifier votre programme pour qu'il gère l'option "**-r**" copiant le contenu d'un répertoire récursivement.