

Système de gestion de fichiers

Appels Système

Manipulation de répertoire

opendir,readdir,closedir,stat

1 test_w

Écrire une fonctionnalité (w) de la commande Unix *test* sur un fichier dont voici une page du manuel.

NOM

`test` - Vérifier le type d'un fichier, et comparer des valeurs

SYNOPSIS

`test` EXPRESSION

DESCRIPTION

Quitter avec un code de retour déterminé par EXPRESSION

Une expression omise est fausse par défaut. Sinon, EXPRESSION est évaluée comme vraie ou fausse, et détermine la valeur de retour. EXPRESSION doit être d'une des formes suivantes :

`-r FICHIER`

FICHIER existe et est lisible

`-w FICHIER`

FICHIER existe et est accessible en écriture

`-x FICHIER`

FICHIER existe et est exécutable (ou peut être parcouru dans le cas d'un répertoire)

"Se termine avec succès" signifie que la commande retourne 0, et donc 1 en cas d'échec
Pour limiter le programme, on se limitera au mode à w.

Donc on cherche simplement à savoir si celui qui exécute la commande *test* a le droit en écriture sur l'entrée passée en paramètre.

```
test -w ~/.bashrc
echo $?
0
test -w ~/etc/passwd
echo $?
1
```

Rappel Unix : Les droits USER, GROUP, OTHER sur une entrée, le propriétaire ne fait pas partie du groupe GROUP. Le propriétaire ni son groupe ne font pas partie des autres OTHER.

Pour simplifier l'exercice, la commande ne consulte pas les groupes secondaires de l'utilisateur de la commande.

Contrainte : La commande doit être construite sur les appels système *stat* qui lit l'inode attaché à l'entrée testée (et n'essaie donc pas d'ouvrir le fichier). On suppose que l'appel à *stat* ou *lstat* n'échoue que si l'entrée n'existe pas. Les fonctions `geteuid()`, `getegid()`, vous donne le pid et gid de l'utilisateur que lance la commande.

En Python `os.stat`, `os.geteuid()`, `os.getegid()`. Interdit d'utiliser `os.access` (c'est justement ce que l'on veut écrire)

Algo : Si je (celui qui exécute la commande `test`) est le propriétaire du fichier, je dois avoir le droit W USR, sinon, si je ne suis pas le propriétaire du fichier mais que je suis dans le groupe de propriétaire je dois avoir W sur le groupe, sinon le droit W doit être positionné pour les autres.

2 Du (Disk Usage)

Ecrire en C, la commande système *du* qui retourne l'espace disque occupé en Kilo Octet par un répertoire. Le nom est passé en argument de la commande. Cette taille se calcule à partir du nombre de "block"s des inodes des entrées contenues dans ce répertoire et de ses sous répertoires.

Rappel : La taille des blocs pour le champs "nombre de blocs", `st_blocks` de la structure `stat` est de 512 octets.

Attention : ne pas compter les blocs de l'entrée "." (je vous laisse réfléchir pour ".").

Vous devriez avoir le même résultat que la commande `du`

```
du -k nomRepertoire
```

Vous utilisez la fonction *ftw* pour faire le parcours récursif du répertoire.

En Python récupérez le programme `ftw.py` sur la plate-forme qui réalise l'équivalent du `ftw` du C.