

INF3172 : TP1

AZIZ SALAH

DATE LIMITE DE LA REMISE : LE LUNDI 10 FEV. 2014 AVANT 23H59

1. OBJECTIF PÉDAGOGIQUE DU TRAVAIL

Manipulation des notions suivantes :

- Système de fichiers

2. DESCRIPTION DU PROBLÈME

Vous avez à concevoir et à codez un programme nommé `whatSize` qui permet d'afficher les tailles les éléments contenus dans un répertoire. Pour un fichier on affiche sa taille et pour un répertoire la taille de sa sous-hiérarchie incluant le répertoire lui même.

`whatSize --` Affiche les tailles du contenu d'un répertoire

SYNOPSIS

`whatSize <repertoire>`

DESCRIPTION

La commande `whatSize` affiche pour chaque élément dans `<repertoire>` une ligne composée de sa taille en octet et du nom de l'élément.

Si l'élément n'est pas un répertoire (un fichier régulier, un lien symbolique ou autre), sa taille est affichée (identique à la taille donnée par `ls -l`).

Si l'élément est un répertoire, sa taille affichée est sa taille (selon `ls`) plus les tailles des éléments qu'il contient en faisant attention au liens physiques.

Lorsqu'on calcule la taille d'un élément qui est un répertoire, tout lien physique dans sa sous-hiérarchie n'est comptabilisé qu'une seule fois dans le calcul la taille de cet élément.

L'affichage de `whatSize` est ordonné selon un ordre décroissant des tailles puis selon un ordre croissant des noms en cas d'égalité des tailles.

3. QUELQUES TESTS D'EXÉCUTION

3.1. **Test1.** Un cas de test simple.

```
malt>tree rep_test1
```

```
rep_test1
```

```
|-- fich1.txt
```

```
|-- fich2.txt
```

```
|-- lien_symb -> rep1/fich4.txt
```

```
|-- rep1
```

```
    |-- fich3.txt
```

```
    |-- fich4.txt
```

```
1 directory, 5 files
```

```
malt>ls -l rep_test1
```

```
total 12
```

```
-rw-r--r-- 1 usager usagers  50 Oct 27 15:36 fich1.txt
```

```
-rw-r--r-- 1 usager usagers  20 Oct 27 15:36 fich2.txt
```

```
lrwxrwxrwx 1 usager usagers  14 Oct 27 15:38 lien_symb -> rep1/fich4.txt
```

```
drwxr-xr-x 2 usager usagers 4096 Oct 27 15:36 rep1
```

```
malt>ls -l rep_test1/rep1
```

```
total 8
```

```
-rw-r--r-- 1 usager usagers 50 Oct 27 15:36 fich3.txt
```

```
-rw-r--r-- 1 usager usagers 50 Oct 27 15:36 fich4.txt
```

```
malt>./whatSize rep_test1
```

```
4196                                rep1
```

```
50                                fich1.txt
```

```
20                                fich2.txt
```

```
14                                lien_symb
```

3.2. Test2. Un cas de test avec des liens physiques.

```
>tree rep_test2
rep_test2
|--- fich1.txt
|--- fich2.txt
|--- rep1
|   |--- fichier3.txt
|   |--- fichier4.txt
|   |--- lien_physique1
|   |--- rep11
|       |--- lien_physique2
|--- rep2
|   |--- fichier5.txt
|   |--- lien_physique3

3 directories, 8 files
malt>ls -li rep_test2
total 16
28052479 -rw-r--r-- 1 usager usagers  50 Oct 27 18:17 fich1.txt
28052481 -rw-r--r-- 1 usager usagers  20 Oct 27 18:17 fich2.txt
28057612 drwxr-xr-x 3 usager usagers 4096 Oct 27 18:31 rep1
28057613 drwxr-xr-x 2 usager usagers 4096 Oct 27 18:37 rep2
malt>ls -li rep_test2/rep1
total 16
28052482 -rw-r--r-- 1 usager usagers  30 Oct 27 18:29 fichier3.txt
28052483 -rw-r--r-- 1 usager usagers  30 Oct 27 18:30 fichier4.txt
28052484 -rw-r--r-- 3 usager usagers  10 Oct 27 18:31 lien_physique1
28057614 drwxr-xr-x 2 usager usagers 4096 Oct 27 18:33 rep11
malt>ls -li rep_test2/rep1/rep11
total 4
28052484 -rw-r--r-- 3 usager usagers 10 Oct 27 18:31 lien_physique2
malt>ls -li rep_test2/rep2
total 8
28052485 -rw-r--r-- 1 usager usagers 60 Oct 27 18:35 fichier5.txt
28052484 -rw-r--r-- 3 usager usagers 10 Oct 27 18:31 lien_physique3
malt>./whatSize rep_test2
8262                rep1
4166                rep2
50                  fich1.txt
20                  fich2.txt
malt>
```

3.3. Tests avec erreurs. Cas des erreurs.

```
malt>./whatSize rep_test3
Erreur : droits insuffisants
malt>./whatSize xyz
Erreur : fichier ou répertoire inexistant
malt>
```

4. EXIGENCES NON FONCTIONNELLES

4.1. Gestion de l'allocation dynamique. Le programme doit s'adapter dynamiquement avec la taille des données en utilisant l'allocation dynamique. Tout espace alloué dynamiquement doit être libéré dès qu'il n'est plus requis.

4.2. Gestion des erreurs. Seules les erreurs considérées par la fonction `print_erreur` sont à traiter dans votre programme.

4.3. Composition du programme.

- `«util.h»` : définit les prototypes de fonctions utilitaires à utiliser pour l'affichage. Le fichier `«util.h»` ne doit pas être modifié.
- `«util.c»` : est une implémentation complète qui vous fournit les fonctions utilitaires pour l'affichage. Les fonctions fournies dans ce fichier ne doivent pas être modifiées.
- `«whatSize.c»` : comporte la fonction `main` et vos fonctions selon votre conception.
- les informations sur les fichiers et répertoires doivent être obtenues à l'aide des fonctions `stat` et `lstat`.
- La compilation de votre programme `«whatSize.c»` se fait par la commande :
`gcc -Wall -std=c99 whatSize.c util.c -o whatSize`
- La compilation de votre programme ne devrait donner aucun avertissement (warning).
- Votre fichier `«whatSize.c»` doit contenir les noms des auteurs et leurs codes permanents en commentaire.
- L'affichage doit respecter les exemples dans les cas d'utilisation fournis.
- Si votre programme `«whatSize.c»` rencontre une erreur, il doit afficher seulement un message d'erreur avec `print_erreur`.
- Tout affichage doit se faire seulement avec les fonctions fournies :
 - `print_element`
 - `print_erreur`

4.4. Portabilité du programme. Afin de s'assurer de la probabilité de votre programme, celui-ci devrait être compilé et testé sur les serveurs malt et rayon1 (ou rayon2). Le premier serveur roule sur Linux alors que les deux derniers sont des machines Sun.

5. CE QUE VOUS DEVEZ REMETTRE

En équipe d'au plus deux personnes, un membre de l'équipe seulement remettra votre fichier `«whatSize.c»` électroniquement dans Moodle en suivant le lien approprié. Pas de remise en double svp.

6. PONDÉRATION

- Tests de fonctionnement : 60%
- Exigences non fonctionnelles : 20%
- Structure du programme, commentaires, indentation ... : 20%

Remarques importantes :

- **Un programme ne compilant pas se verra attribuer la note 0.**
- Aucun programme reçu par courriel ne sera accepté. En cas de panne des serveurs, un délai supplémentaire vous sera accordé sans pénalité de retard.
- Les règlements sur le plagiat seront strictement appliqués.
- 10% comme pénalité de retard par journée entamée. Après cinq jours, le travail ne sera pas accepté.
- La remise d'un fichier "zipé" donne lieu à 10% de pénalité.