



GU : Licence Sciences Technologies (semestre 2)
Épreuve de : Utilisation des systèmes informatiques
Date : 3 septembre 2004 - Saint Grégoire
Documents manuscrits autorisés
Épreuve de Monsieur Delmas

UE : INF103

Durée : 1h30

Répondre dans les cadres directement sur le sujet ou cocher lorsque nécessaire la ou les bonnes réponses - ATTENTION : cocher une mauvaise réponse retire un point
Indiquer votre identificateur d'anonymat :

* Peut-on faire fonctionner des programmes sur un ordinateur en l'absence de système d'exploitation ?

Oui Non

* Décrire brièvement le rôle d'un système d'exploitation

[Empty box for describing the role of an operating system]

* Cochez les systèmes d'exploitation qui sont du type Unix.

Sun/Solaris Linux
MacOS X Windows NT
NetBSD FreeBSD

* Sur les systèmes de type Unix citez :

3 programmes clients de lecture des forums de discussion
3 programmes clients de lecture du courrier électronique
3 programmes clients de navigation web

* Quel est le nom du fournisseur d'accès Internet de l'Université Bordeaux 1 ?

free renater
aol reamur
tiscaly wanadoo

* Comment faire afficher par le shell bash la valeur de retour d'une commande ?

echo \$\$ echo ?
echo # echo \$!
echo \$* echo \$?

* Cochez les affirmations vraies : le super utilisateur NE peut PAS,

- | | | | |
|--|--------------------------|---|--------------------------|
| changer le mot de passe des utilisateurs | <input type="checkbox"/> | traverser tous les répertoires | <input type="checkbox"/> |
| changer les droits des fichiers des utilisateurs | <input type="checkbox"/> | connaître le mot de passe des utilisateurs | <input type="checkbox"/> |
| lire le contenu de tous les fichiers utilisateurs | <input type="checkbox"/> | retrouver un fichier utilisateur effacé par mégarde | <input type="checkbox"/> |
| changer les liens de parenté entre processus | <input type="checkbox"/> | changer un compte utilisateur en super utilisateur | <input type="checkbox"/> |

★ L'utilisateur login-1 du groupe unix etu exécute dans son *shell* les commandes suivantes :

```
$ cd
$ chmod 755 .
$ mkdir perso
$ chmod 732 perso
$ cat > perso/resultat.txt
Hello, world.
^d
$
```

Cochez les commandes pouvant être effectuées sans erreur par un utilisateur login-2 du groupe unix etu.

- | | | | |
|---|--------------------------|--|--------------------------|
| \$ ls -al ~login-1/perso | <input type="checkbox"/> | \$ cat > ~login-1/perso/resultat-2.txt | <input type="checkbox"/> |
| \$ ls ~login-1/perso/resultat.txt | <input type="checkbox"/> | 2002 Gödel Prize - Geraud Senizergues | <input type="checkbox"/> |
| \$ cd ~login-1/perso | <input type="checkbox"/> | ^d | <input type="checkbox"/> |
| \$ rm ~login-1/perso/resultat.txt | <input type="checkbox"/> | \$ mkdir ~login-1/perso/projet | <input type="checkbox"/> |

★ Écrire le résultat produit par la suite de commandes :

```
$ cd ; pwd
/net/usr/login
$ FOO=pwd
$ N="(cd ~/.; (cd ../.././.; pwd); eval $FOO); pwd"
$ eval $N; eval $N; eval $N; pwd
```

★ Écrire le résultat produit par la suite de commandes :

```
$ cd ; pwd
/net/usr/login
$ N="cd .; (cd ~/.; (cd ../.././.; pwd) ; pwd ); pwd"
$ eval $N; (eval $N; (eval $N; (eval $N))); pwd
```

★ Que produit l'exécution des commandes suivantes

```
$ ls /etc
ssh ntp ynu yp yn.conf yn.rpmnew ynu.conf ynu.canz ynu.rpnew zsh
$ N='/etc/[y-z]??.*[f-w]'
```

```
$ echo "$M"
```

★ Que produit l'exécution des commandes suivantes

```
$ HELLO=world
$ sh
$ echo /$HELLO/
```

```
$ ^d
$
$ ( export TMP=/tmp ); export FOO=bar
$ bash
$ export TREIZE=XIII
$ echo /$HELLO/$TMP/$FOO/$TREIZE/
```

```
$ ^d
$ echo \*$TREIZE\*
```

* Que produit l'exécution des commandes suivantes

```
$ echo "main (void) { printf("printf peut afficher ceci. \n"); }" | tr -cs [:alnum:] '\n'
```

* Écrire un script-shell `frequence-des-mots.sh` qui se comporte comme un filtre et qui affiche sur la sortie standard la fréquence d'apparition (classé par ordre décroissant) de chaque mot ou nombre provenant du flux d'entrée standard. Par exemple,

```
$ cat essai.c
main (int argc, char **argv)
{
    printf("Hello, world - Hello everybody \n");
    exit(0);
}
$ ./statistique.sh < essai.c
    2 Hello
    1 world
    1 printf
    1 n
    1 main
    1 int
    1 exit
    1 everybody
    1 char
    1 argv
    1 argc
    1 0
$
```

- * Écrire un script-shell `change-droits-acces-repertoires.sh` qui prend en paramètre un nom de répertoire et une chaîne de caractères correspondant à des droits d'accès. Ce script doit positionner les droits d'accès (fourni par le second paramètre) sur le répertoire (fourni par le premier paramètre) ainsi que tous ces sous-répertoires, le script ne devra afficher aucun message d'erreur. Le script devra fonctionner même s'il y a des répertoires commençant par le caractère point.

- * Expliquer ce qui différencie un programme exécutable d'un processus.

- * Quel est l'ordre de création des processus lorsque l'on tape la commande :

```
$ date | cut -d ' ' -f4 | tr ':' '-'
```

- `date` puis `cut -d ' ' -f4` puis `tr ':' '-'`
- `tr ':' '-'` puis `cut -d ' ' -f4` puis `date`
- `cut -d ' ' -f4` puis `date` puis `tr ':' '-'`

`date` et `cut -d ' ' -f4` et `tr ':' '-'` sont créés en même temps

- * Comment peut-on passer un processus en avant-plan dans l'état STOP ?

Control-y Escape
Control-z Control-d

★ Quel est le PID du noyau d'un système de type UNIX ?

-1 1
0 2

★ Le script-shell mystere.sh prend 2 paramètres, un entier suivi d'un nom de fichier.

```
$ cat mystere.sh
#!/bin/bash

if [[ ( $(wc -l < $2) -lt $1 ) || ( $1 -le 0 ) ]]
then
  cat $2
else
  cat $2 | head -$(expr $1 - 1)
  cat $2 | tail -$(expr $(wc -l < $2) - $1)
fi
$
```

En supposant que le fichier debarquement.txt contienne

```
$ cat debarquement.txt
Les sanglots longs
Des violons
De l'automne
Blessent mon coeur
D'une langueur
Monotone.
$
```

Que produira la commande

```
$ ./mystere.sh 10 poeme.txt; ./mystere.sh 3 poeme.txt
```

★ Le script-shell mkdir-cd.sh a pour but de créer un nouveau répertoire et de placer l'utilisateur directement dans ce répertoire

```
$ cat mkdir-cd.sh
#!/bin/bash

mkdir $1
```

```
cd $1
$
```

Donner un exemple d'utilisation de ce script `mkdir-cd.sh` sous votre shell afin de créer et de vous placer directement dans le répertoire `~/foo` (on supposera que ce répertoire n'existait pas).

★ Écrire une autre solution permettant à l'utilisateur d'utiliser une commande `mkdir-cd` mais qui n'utilise pas de script-shell.

★ Écrire un script shell `enlever-des-lignes.sh` qui prend en premier paramètre un nom de fichier puis une liste d'entier. Ce script doit renvoyer sur la sortie standard le contenu du fichier passé en premier paramètre auquel on aura supprimé les lignes passées via les autres paramètres. Par exemple,

```
$ ./supprime-ligne fichier.txt 2 5 11
```

devra renvoyer le contenu du fichier `fichier.txt` supprimé des lignes 2, 5 et 11.

Votre script devra supporter n'importe quel ordre de passage des numéros de lignes. Par exemple,

```
$ ./supprime-ligne fichier.txt 11 2 5
```

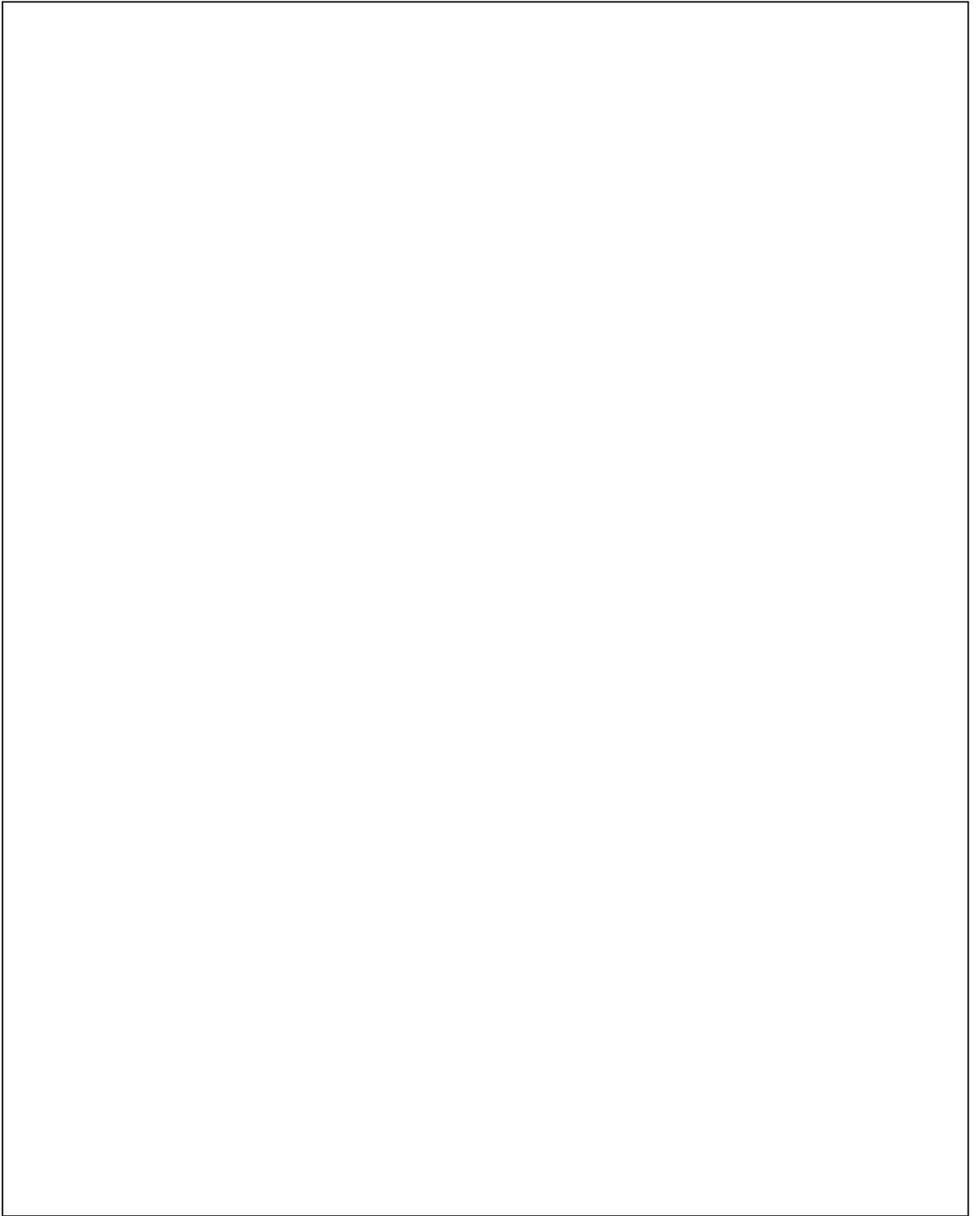
 aura le même effet.

Votre script devra également prendre en compte des intervalles de lignes via la notation `<numéro-début-ligne>-<numéro-fin-ligne>` (avec `numero-debut-ligne` toujours inférieur ou égal à `numero-fin-ligne`). Par exemple

```
$ ./supprime-ligne fichier.txt 3-5 1
```

 aura pour but de supprimer les lignes n° 1, 3, 4 et 5.

Si un numéro de ligne passé en paramètre n'existe pas dans le fichier, cela n'affectera pas le contenu du fichier renvoyé par le script. Par contre votre script devra effectuer les vérifications nécessaires d'existence du fichier ainsi que la vérification des types (entier positif, ou intervalle) passer en paramètres après le nom du fichier. Votre script devra renvoyer un message d'usage sur la sortie d'erreur standard et positionner une valeur de retour adéquate.



FIN.