



Sans documents, 1h30.

1 Droits d'accès

La commande `ls -l` montre ceci :

```
-rwxr----- 1 michel profs 12151 mar 1 15:31 examen.doc
-rwxr----- 1 michel profs 18853 mar 1 15:45 solutions.doc
-rwx----- 1 michel profs 2344 jan 23 8:12 essai.sh
```

- 1 Qui peut lire le fichier `examen.doc` ?
- 2 Comment modifier les droits d'accès pour que les autres professeurs (membres du groupe `profs`) puissent modifier le fichier `solutions.doc` ?
- 3 Après la commande
`chmod 751 essai.sh`
qui peut faire exécuter ce script ?

2 Chemins

- 4 Expliquez ce qu'est un *chemin d'accès relatif*.
- 5 Complétez la table de correspondance entre chemins.

| Répertoire de travail | Chemin d'accès relatif | Chemin absolu |
|------------------------|------------------------|---------------------------|
| /etc | apache2/conf.d | (a) |
| (b) | bin/sauvegarde | /usr/local/bin/sauvegarde |
| /usr/local/doc/systeme | (c) | /usr/local/bin/sauvegarde |

3 Écriture d'un script

- 6 Écrire une commande shell `iota` à un paramètre entier n , qui affichera la suite des entiers de 1 à n

Exemple d'exécution de cette commande

```
$ iota
Erreur: il faut un parametre
$ iota 4
1
2
3
4
```

Rappel : la variable « `##` » comporte le nombre de paramètres passés à l'appel d'une procédure de commande.

4 Lecture de script

Voici le texte d'un shell-script `mkp.sh`

```
1  #!/bin/bash
2
3  function mk-header
4  {
5      maj=$(echo $1 | tr a-z A-Z)
6      cat <<EOF
7  #ifndef ${maj}_H
8  #define ${maj}_H
9  class $1 {
10     public:
11         $1();
12         ~$1();
13     };
14 #endif
15 EOF
16 }
17
18 function mk-code
19 {
20     cat <<EOF
21     #include "$1.h"
22     $1::$$1() {
23     }
24     $1::~~$1() {
25     }
26 EOF
27 }
28
29 function mk-prog
30 {
31     cat <<EOF
32     #include <iostream>
33     using namespace std;
34     int main () {
35     }
36 EOF
37 }
38
39 for nom in $*
40 do
41     case $nom in
42         *cc)
43             mk-prog $nom > $1
44             ;;
45         *)
46             mk-header $nom > $1.h
47             mk-code $nom > $1.cc
48             ;;
49     esac
50 done
```

7 Expliquez en détail le déroulement de la commande

```
$ ./mkp.sh prog.cc
```

Indiquez précisément

- les numéros de lignes de script qui sont exécutées
- le nom et le contenu des fichiers produits

8 Même question pour la commande

```
$ ./mkp.sh Polygone
```

9 À quoi sert la boucle `for` de la ligne 39 ? Donnez un exemple d'utilisation.

10 Expliquez le déroulement de la commande de la ligne 5

```
maj=$(echo $1 | tr a-z A-Z)
```

quand le script est lancé avec le paramètre `Polygone`.

5 Envoi des résultats d'un examen

Rappel : la commande `mail` permet d'expédier un courriel lu sur l'entrée standard. Usage :

```
mail [-s sujet] adresse...
```

5.1 Étude d'un script

Le script suivant, appelé `envoi.sh`, est destiné à envoyer une “*newsletter*” à un ensemble d'abonnés.

```
#!/bin/bash

#
# Ce script envoie la lettre d'informations aux lecteurs
#

D=$(date +%d %B %Y')
mail -s "Lettre d'information du $D" $(cat $1) < $2
```

11 Quels paramètres faut-il donner à ce script, et sous quelle forme ? (adresse mail, nom d'un fichier, message, chaîne de caractère du message, sujet, fichier d'adresses, ... ?)

Donnez un exemple concret d'utilisation de ce script.

5.2 Production d'un fichier d'adresses

On dispose d'un fichier `abonnes.csv` d'abonnés de la forme

```
marcel,proust,mp@yagoo.com
victor,hugo,totor@snail.fr
francois,rabelais,gargantua@theleme.fr
...
```

avec les adresses de courrier électronique en troisième position (les champs sont séparés par des virgules).

12 Écrivez une commande qui produit un fichier d'adresses à partir d'`abonnes.csv`.

5.3 Envoi des notes d'un devoir

Le fichier `notes-ds.txt` suivant contient les notes d'un devoir de 4 questions de l'université de Bordeaux.

```
albert.dupont:3:4:2:5  
bruno.durand:1:4:3:3  
..
```

13 Écrivez un script qui prendra ce fichier en paramètre, et expédiera un courrier à chaque étudiant.

Par exemple, le message

```
Sujet: Note devoir = 11  
Vous avez au 1, 4, 3 et 3 aux questions du devoir de samedi,  
soit 11 sur 19 au total.  
sera envoyé à bruno.durand@etu.u-bordeaux-1.fr.
```

Remarque : tous les étudiants de l'université Bordeaux 1 ont une adresse de la forme

`nom.prenom@etu.u-bordeaux1.fr`