

EXAMEN COURT DE SYSTÈME D'EXPLOITATION

IUT DE VILLETANEUSE, DÉPARTEMENT INFORMATIQUE

Votre nom et votre prénom :

Votre groupe :

Exercice I : Interprétation d'un code Bash (10pts)

Soit le script shell suivant :

```
# "g" - Go to bookmark
function g () {
    if [ -z "$1" ]
    then
        cd
    else
        if [ -f ~/.DirB/"$1" -a ${1:0:1} != "." ]
        then
            touch ~/.DirB/"$1" ;
            cd 'cat ~/.DirB/"$1"' ;
        else
            cd "$1"
        fi
    fi
}

# "r" - Remove a saved bookmark
function r () {
    if [ -e ~/.DirB/"$1" ]
    then
        rm ~/.DirB/"$1"
    else
        echo bash: DirB: ~/.DirB/"$1" does not exist >&2
        false
    fi
}
```

Date: Mardi 2 novembre 2010, 09:30-11:00.

```
# "p" - Push a bookmark
function p () {
  if [ -z "$1" ]
  then
    pushd > /dev/null && dirs -p
  elif [ "$1" == "-" ]
  then
    popd > /dev/null
    dirs -p
  else
    if [ -f ~/.DirB/"$1" -a "${1:0:1}" != "." ]
    then
      touch ~/.DirB/$1 ;
      pushd 'cat ~/.DirB/$1' > /dev/null && dirs -p ;
    else
      pushd "$1" > /dev/null && dirs -p
    fi
  fi
}
```

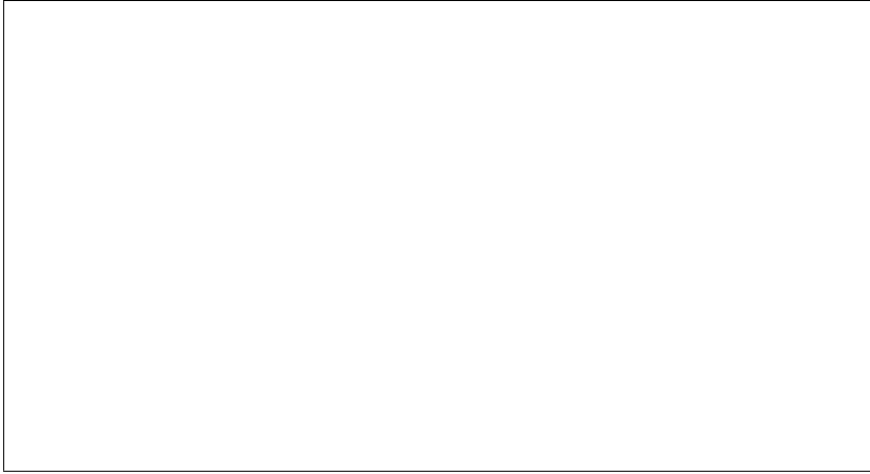
On rappelle que :

- **pushd** : Sauvegarde et change le répertoire courant
- **popd** : Restaure le dernier répertoire sauvegardé par pushd
- **dirs** : Affiche la liste des répertoires sauvegardés

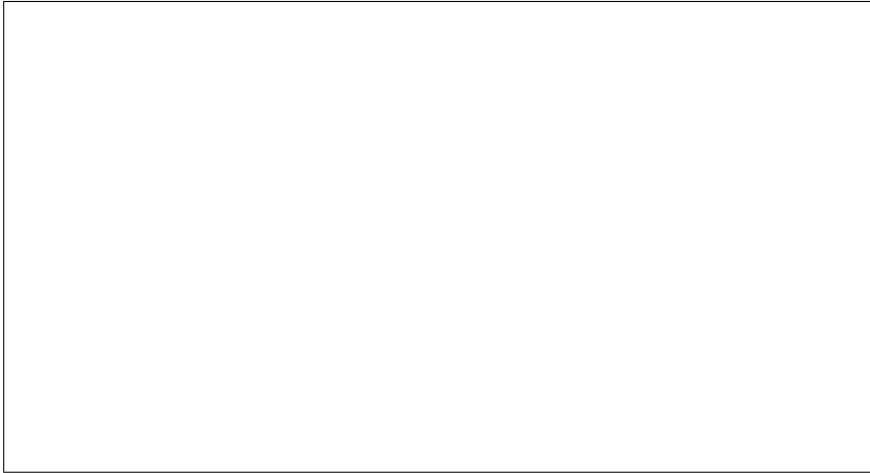
Question 1.1. *Expliquer le code des fonctions suivantes. Comment s'utilisent-elles ?*

1.1.1. *fonction g ()*

1.1.2. *fonction r* ()

A large, empty rectangular box with a thin black border, intended for the student to write their answer to question 1.1.2.

1.1.3. *fonction p* ()

A large, empty rectangular box with a thin black border, intended for the student to write their answer to question 1.1.3.

Exercice II : Construction d'un code Bash (6pts)

Question 2.1. *On vous demande de compléter le code source précédent en écrivant la fonction `sl` qui va afficher la liste des bookmarks sauvegardés dans `~/DirB/` et le contenu des bookmarks sauvegardés.*

Exercice III : Expression régulière (4pts)

La commande `cal` retourne le calendrier d'un mois donné. Par exemple pour le prochain mois, la commande `cal 11 2010` retourne :

```
Novembre 2010
lu ma me je ve sa di
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
29 30
```

Question 3.1. Donner une expression régulière '*REG*' qui va permettre de récupérer les semaines entières dans le mois. Ainsi

```
cal 11 2010 | grep -E 'REG'
```

affichera :

```
 1  2  3  4  5  6  7
 8  9 10 11 12 13 14
15 16 17 18 19 20 21
22 23 24 25 26 27 28
```

Veillez écrire votre réponse ci dessous en expliquant pourquoi vous êtes arrivés à cette expression régulière.

Question 3.2. Expliquer pourquoi

```
cal 11 2010 | grep -E "[[:digit:]]+[[:blank:]]*"
ne fonctionne pas.
```



Propositions de corrections

#Exercice 1 :

```
: << '--Question 1--'
(4 pts)
"g" - Go to bookmark
```

Utilisation : g [<bookmark> | <repertoire>] (1pt)

Description :

g : (tout seul) change repertoire courant (PWD) par le repertoire d'accueil (HOME) (1pt)

g <bookmark> : (<bookmark> est un fichier ordinaire non caché contenant un chemin de repertoire) change le repertoire courant (PWD) par le repertoire contenu dans <fichier> (1pt)

g <repertoire> : change le repertoire courant (PWD) par le repertoire <repertoire> (1pt)

--Question 1--

```
function g () {
  # test chaine vide
  if [ -z "$1" ]
  then
    cd
  else
    # $1 est un fichier non caché© contenant un chemin
    if [ -f ~/.DirB/"$1" -a ${1:0:1} != "." ]
    then
      #
      # estampille $1
      # cd chemin
      touch ~/.DirB/"$1" ;
      cd 'cat ~/.DirB/"$1"' ;
    # $1 repertoire (pas fichier ordinaire ou commence par .
    else
      cd "$1"
    fi
  fi
}
```

: << '--Question 2--'

(1 pt)

"r" - Remove a saved bookmark

Utilisation : r <bookmark>

Description :

- supprime le fichier <bookmark> s'il existe
- sinon affiche un message d'erreur

--Question 2--

```
function r () {
    # supprime $1
    if [ -e ~/.DirB/"$1" ]
    then
        rm ~/.DirB/"$1"
    # message erreur
    else
        echo bash: DirB: ~/.DirB/"$1" does not exist >&2
        false
    fi
}
```

: << '--Question 3--'

(5 pts)

"p" - Push a bookmark

Utilisation : p [- <bookmark> <repertoire>] (1pt)

Description :

- p : p (seul) permute les 2 repertoires en haut de pile puis cd. affiche contenu pile (1 par ligne) (1pt)
- p - : depile et cd. Affiche contenu pile (1 par ligne) (1pt)
- p <bookmark> : (<bookmark> est un fichier ordinaire non caché contenant un chemin de repertoire).
p <bookmark> - estampille le fichier <bookmark>

- empile le chemin du <bookmark> puis cd
- affiche contenu pile (1 par ligne) (1pt)

```
p <repertoire> :  
    - empile le chemin du <bookmark> puis cd  
    - affiche contenu pile (1 par ligne) (1pt)
```

--Question 3--

```
function p () {  
    if [ -z "$1" ]  
    then  
        pushd > /dev/null && dirs -p  
    elif [ "$1" == "-" ]  
    then  
        popd > /dev/null  
        dirs -p  
    else  
        if [ -f ~/.DirB/"$1" -a "${1:0:1}" != "." ]  
        then  
            touch ~/.DirB/$1 ;  
            pushd 'cat ~/.DirB/$1' > /dev/null && dirs -p ;  
        else  
            pushd "$1" > /dev/null && dirs -p  
        fi  
    fi  
}
```

#Exercice 2 :

"sl" - affiche liste des bookmarks et contenu

```
function sl () {  
    echo -e "Bookmark \t Contenu"  
    for i in `ls ~/.DirB` (2 pts boucle)  
    do  
        if [ -f "$i" ] (2 pts test)  
        then  
            echo -e "$i \t 'cat ~/.DirB/$i'" (2pts echo)  
        fi  
    done  
}
```

```
# Exercice 3 :
```

```
# Question 1
```

```
# (2pts)
```

```
cal 11 2010 | grep -E '^[ 0-9][0-9]( [ 0-9][0-9]){6}$'
```

```
cal 11 2010 | grep -E
```

```
'^[[[:blank:]][:digit:]][:digit:]]( [[[:blank:]][:digit:]][:digit:]]){6}$'
```

```
# Question 2
```

```
: << '--Question 2--'
```

```
(2pts)
```

```
l'expression d'écrit une ligne du calendrier comme :  
  une suite (éventuellement vide)  
  de chiffre (au moins 1) mais sans limite  
  separees par 0 ou plusieurs blancs
```

```
--Question 2--
```