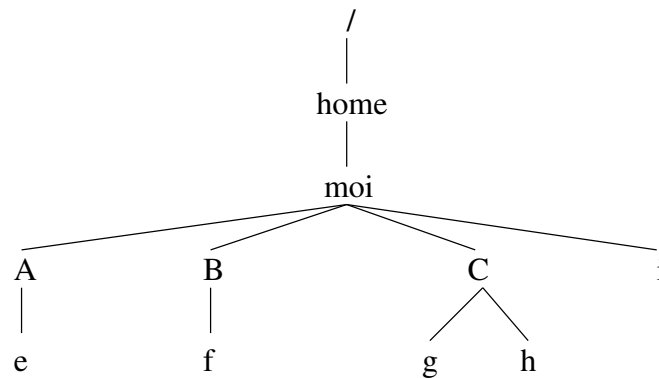


L1*Examen de système*Bernard GOOSSENS [bernard.goossens@univ-perp.fr]

- *La durée de l'examen est de 2h00.*
- *L'accès à une machine et à tous les documents est autorisé.*
- *Seules les réponses dans les rectangles prévus à cette effet seront considérées.*

EXERCICE 1 **Fichiers****FIGURE 1** – Arborescence de dossiers et de fichiers

Soit l'arborescence de la figure 1. On suppose que le dossier courant est le dossier personnel *moi*.

1. Donner la suite de commandes nécessaires (`cd`) pour effectuer les opérations suivantes. Descendre dans le dossier *A*. Se déplacer au dossier *B*. Remonter au dossier *home* (par un chemin relatif). Se placer sur le dossier *C* (par son chemin absolu). Revenir au dossier *moi*.

2. Comment employer la commande `cat` pour afficher les fichiers *i* et *h* sans changer de dossier courant ?

- Copier avec la commande `cp` le fichier *f* dans le dossier *C* depuis le dossier *moi*.
Changer la copie de nom pour la renommer *j* (commande `mv`), toujours depuis le dossier *moi*.

- Créer un dossier *D* (`mkdir`) dans le dossier *moi*. Changer les droits d'accès à *D* (`chmod`) pour autoriser les membres du groupe propriétaire à créer des fichiers dans *D*.

- Créer un lien *l* (commande `ln`) dans le dossier *D* sur le fichier *g*. Supprimer la référence *g* (commande `rm`). Restaurer la référence *g* dans *C* en utilisant la commande `ln`.

- Créer avec la commande `echo` un fichier *m* dans le dossier *A* contenant le texte "bonjour". Ajouter la ligne "au revoir" au fichier *m* avec la même commande `echo`. Avec la commande `cat`, créer un fichier *n* dans le dossier *A* contenant deux fois le fichier *m*.

EXERCICE 2 Processus

F	S	UID	PID	PPID	C	PRI	NI	ADDR	SZ	WCHAN	TTY	TIME	CMD
4	S	0	1	0	0	80	0	-	6840	poll_s	?	0:01	/sbin/init
4	S	0	1885	1	0	80	0	-	68907	poll_s	?	0:00	lightdm
4	S	0	2425	1885	0	80	0	-	48197	wait	?	0:00	lightdm
4	S	1000	2802	2425	0	80	0	-	7684	poll_s	?	00:00:06	init
0	S	1000	3697	2802	0	80	0	-	185859	poll_s	?	00:00:08	gnome-terminal
0	S	1000	3707	3697	0	80	0	-	6056	wait	pts/0	00:00:00	bash
0	S	1000	3769	3707	0	80	0	-	13833	poll_s	pts/0	00:00:01	xfig
0	R	1000	12682	3707	0	80	0	-	2817	-	pts/0	00:00:00	ps
0	S	1000	12008	2802	0	90	10	-	199105	poll_s	?	00:00:11	update-manager
0	S	1000	10688	3697	0	80	0	-	6052	n_tty_	pts/2	00:00:00	bash

La figure ci-dessus représente une partie de l'affichage de la commande `ps -e1`. Il y a deux terminaux ouverts, chacun exécutant un *shell bash*. Le premier est nommé `pts/0` et le second est nommé `pts/2`.

1. Dessiner l'arbre généalogique des processus.

2. Sur quel terminal a été exécutée la commande `ps -e1` ?

3. Si on exécute la commande `sh` dans le terminal `pts/2` quel sera son PPID ?

4. Comment fermer le terminal `pts/2` depuis le terminal `pts/0` (commande `kill -9`) ? Depuis `pts/2` lui-même (donner deux réponses différentes) ?

EXERCICE 3 Scripts

```
#!/bin/bash
C=$1
echo $C
F=$((18*$C)/10+32))
echo $F
```

Soit le script de la figure ci-dessus qui convertit en degré Fahrenheit une température C en degré Celsius.

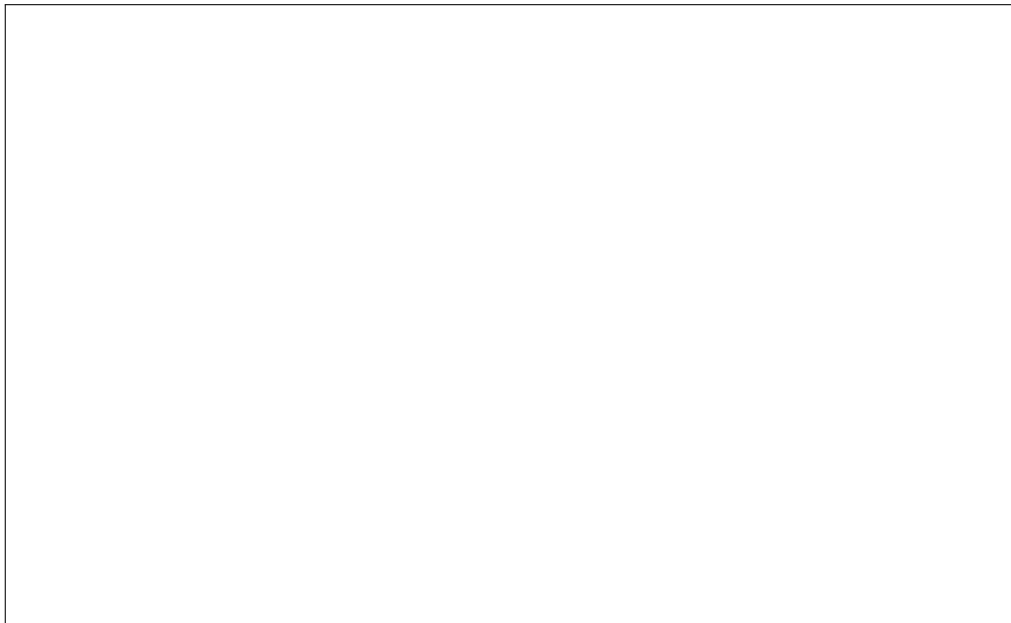
1. A quoi correspond "\$1" ?

2. Ecrire un script faisant la conversion inverse (formule : $C = (5F-160)/9$).

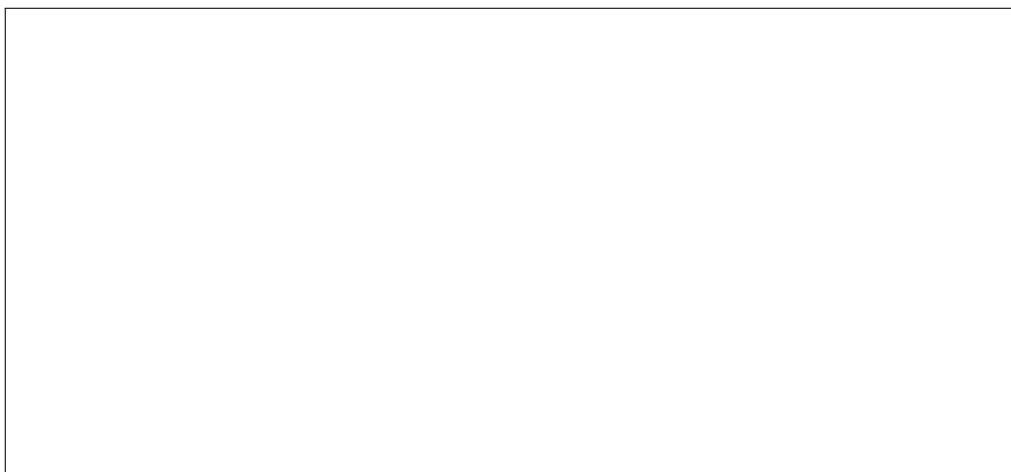
```
#!/bin/bash
A=$1
B=$2
if [ $A -gt $B ];
then
    echo $A
else
    echo $B
fi
```

Soit le script de la figure ci-dessus qui affiche le plus grand parmi deux entiers A et B . Le script est appelé `sup2`.

3. Sur le modèle de `sup2`, écrire un script pour afficher le plus grand parmi trois entiers A , B et C .



4. Ecrire une nouvelle version sans *if* en utilisant `sup2` (utiliser l'évaluation de commande : '`commande paramètres`' retourne la chaîne de sortie de `commande` ; par exemple '`echo toto`' retourne "`toto`" ; `sup3(A,B,C)=sup2(sup2(A,B), sup2(B,C))`).



```
#!/bin/bash
N=$1
for i in $(seq 0 $N)
do
    printf "%s " $i
done
printf "\n"
```

Soit le script `liste` de la figure ci-dessus qui affiche sur une même ligne les N premiers entiers.

5. Sur le modèle de `liste` écrire un script `carre` qui affiche les N premiers carrés sans utiliser de multiplication (tout carré est une somme d'entiers impairs : par exemple, $16 = 1 + 3 + 5 + 7$).

