

Examen court de Système d'Exploitation

Département Informatique IUT Villetaneuse, Semestre décalé

Durée 70 minutes

1- Commandes unix (7 pts, 10min)

1. Sur les machines de l'IUT, pour mettre dans votre répertoire racine l'exécutable /bin/bash il suffit de faire a) `mv /bin/bash ~` b) `cp /bin/bash ~` c) on peut faire a) ou b), au choix. Vous justifierez votre réponse.
2. A votre avis, pourquoi obtient on le résultat suivant à l'exécution de :

```
cerin@linux:> echo "$#" $1 $2 3
0 3
```
3. Donnez un exemple de boucle (`for` ou `while`) écrite en bash qui ne termine pas.
4. Quel va être le résultat de la commande suivante ? Justifiez votre réponse.

```
j=12;case "$j" in [0-9][0-9] ) echo "double digit";; * ) echo "No";;
esac
```
5. Quel va être le résultat (à priori) de l'exécution des commandes suivantes ? Vous justifierez votre réponse :

```
cerin@linux:~> j=`pwd`;if [ j == "/home/cerin" ]; then echo "Vrai";
else echo "Faux"; fi
```
6. La commande `grep` permet de rechercher un motif sur chacune des lignes d'un fichier ? Vrai ou faux ?
7. Si le fichier `ara.txt` contient sur chaque ligne le nom et le prénom des étudiants du semestre décalé, quel va être le résultat de la commande suivante :

```
cerin@linux:~> grep -G [[:digit:]] ara.txt
```

Justifiez votre réponse.

2- Scripts

2.1 Dessin (5 pts - 15min)

Ecrire un script qui se comporte de la manière suivante :

```
cerin@taipei:$ /bin/sh test.bash
Enter Number (< 10): 6
 6
 5
4 6
3 5
2 4 6
1 3 5
 2 4 6
 3 5
 4 6
 5
 6
```

Notes : attention à la présentation désirée. Il y a certainement des boucles dans le programme ! Ne vous dispensez pas d'explications !

Quels (petits) problèmes de présentation du dessin va t-il se poser si l'entier saisi est plus grand que 10 ? Donner quelques éléments/explications pour régler le problème... et, si vous pouvez, le code correspondant (ce code correspond à une question facultative qui ne peut que vous donner 1 point supplémentaire... mais ne perdez pas de temps sur cette mini-question).

2.2 Courbe (8 pts – 45min)

Ecrire un script qui prend en paramètre une fonction trigonométrique ($3*\cos(x)$) dans notre premier exemple ci dessous, et qui trace cette fonction dans une fenêtre `xterm`.

Dans le premier exemple, il s'agit de tracer $f(x)=5*\cos(1.5*x)$ entre 0.0 et 1.0 par pas de 0.1 (x évolue de 0.1 en 0.1 depuis la valeur initiale 0.0).

a) Vous pouvez utiliser la commande `bc` pour calculer $f(x)$ mais elle ne reconnaît que `c()` pour la fonction cosinus. Il faudra donc substituer `cos` par `c` dans la chaîne passée en paramètre. Quelle commande Bash allez vous écrire pour ranger dans la variable `f` le résultat de la substitution de `cos` par `c` dans la chaîne `$1` ? **(1pt)**

b) Bash ne sait pas faire de l'arithmétique sur les réels : il faudra également passer par la commande `bc -l` car les bornes des itérations et l'incrément (`$2`, `$3`, `$4`) sont des réels. Donnez l'expression Bash correspondant au critère d'arrêt des itérations. **(1pt)**

c) Note technique : concernant le dessin dans une fenêtre texte, vous pouvez utiliser les instructions suivantes :

```
echo -e -n "\033[7s" #save current screen postion & attributs
tput cup 0 69      # row 0 and column 69 is used to show a star
echo -n "*"        # put a star on screen
echo -e -n "\033[8u" #restore current screen postion & attributs
```

N'oubliez pas d'expliquer ce que vous faites, de donner par exemple l'algorithme que vous suivez **(2 pts)**. Les **4 pts** qui restent sont distribués pour rassembler les différentes idées.

L'écran devrait rassembler à ceci après l'exécution des programmes, programmes lancés de la façon suivante dans une console `xterm` :

```
cerin@taipei:~$ ./test1.bash "5*cos(1.5*x)" 0.0 1.0 0.1
```

```
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
```

```
cerin@taipei:~$ ./test1.bash "5*x*cos(1.5*x)" 0.0 1.0 0.1
```

```
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
*
```

Note : l'orientation des axes n'est pas habituelle ici mais c'est secondaire.