

Classe:.....

Nom:.....

Groupe:

Prénom:.....

**CONTRÔLE : SYSTÈME D'EXPLOITATION 4
3IIR**

Barème : 2 points par question

A- Soit le crontab de l'utilisateur (emsi) qui contient les trois lignes suivantes :

```
MAILTO=admin@emsi.ma
```

```
0 0 * * 1 /usr/local/myapp/script1 | grep "error"
```

```
0 0 * * 5 /usr/local/myapp/script1 | grep "error"
```

1. Expliquer à quels moments ce script sera-t-il exécuté?

tous les lundis et les vendredis à minuit

.....
.....

2. Réécrire les deux tâches Cron en une seule ligne

```
0 0 * * 1,5 /usr/local/myapp/script1 | grep "error"
```

.....
.....

3. Le service Cron enverra-t-il quelque chose à l'adresse email mentionnée? **OUI**. Si OUI, quoi? Si NON, pourquoi?

toutes la sortie d'erreur du script en plus des lignes qui contiennent la chaîne "error" de la sortie principale

.....
.....

B- Soit un script shell appelé de la manière suivante :

```
./monscript.sh bonjour tout le monde
```

1. La première chose que nous souhaitons faire dans ce script c'est de vérifier si \$1 est égale à "bonjour", dans ce cas on l'écrase par "salut". Donnez l'expression complète pour la condition et l'écrasement.

```
[ "$1" = "Bonjour" ] && set Salut $2 $3 $4
```

.....
.....

2. Le script "monscript.sh" contient la commande suivante:

```
echo ${4:-"peuple"}
```

Quel est le résultat affiché par cette commande ?

monde

3. Quel serait le résultat affiché par la commande suivante si elle est ajoutée à la fin de notre script monscript.sh ?

```
echo ${2:2}
```

ut

C- Quel est le raccourci clavier qui permet d'envoyer un signal kill -2 à la commande en cours d'exécution?

Ctrl + C

D- On a une variable PS2=">" déclarée dans le fichier ~/.profile et PS2=">>" dans ~/.bashrc . Laquelle des deux valeurs on va trouver si on ouvre un terminal et on tape la commande suivante : echo \$PS2 ? >>

Pourquoi?

parce que .bashrc s'exécute après .profile; donc la valeur de la variable sera écrasée

E- Soit la commande suivante tapée par l'utilisateur "belatar" dans un shell bash :

```
PS1="\[\e[31m\]Hi \u $ " && echo "bonjour"
```

1. Quel serait le résultat final affiché :

bonjour

Hi belatar \$ (le prompt sera affiché en couleur)

.....
2. Il y a un problème dans cette commande, trouvez le problème et réécrivez-la pour remédier à ce problème :

Tout le texte qui vient après le prompt sera affiché en couleur. Il faut revenir à la couleur par défaut :

```
PS1="\[\e[31m\]Hi \u $ \[\e[00m\]"
```

.....
.....