

Classe:.....

Nom:.....

Groupe:

Prénom:.....

CONTRÔLE : SYSTÈME D'EXPLOITATION 4

3IIR

1^{ère} partie : compréhension

A- Que fait la commande "env"? (1pt)

affiche toutes les variables d'environnement

B- Citez trois situations où l'on peut avoir besoin d'utiliser des scripts shell. (3pt)

1. **tâches répétitives nécessitant plusieurs commandes/options**
2. **tâches planifiées automatisées**
1. **opérations d'installation/configuration de logiciels**

C- Citez deux moyens pour modifier la valeur des variables positionnelles (\$1, \$2, \$3...) dans un script bash? (2pt)

1. **set x y z**
2. **shift N**

D- Quand il y a une erreur, on fait souvent appel à la commande "exit" :

1. Que fait la commande "exit 1" ? (2pt)

elle permet de quitter le script sans aller jusqu'à la fin en retournant le code d'erreur 1 au shell

2. Peut-on lui donner une valeur différente de 1? (2pt)

OUI NON

2^{ème} partie : interprétation

A- Quel serait le résultat à l'affichage du script suivant: (4pt)

```
#!/bin/bash
var3=15
var1=`(var2=15;var3=$(( $var2+5 )); var3=$(( $var3*2 )); echo $var3)`
echo $var1
var4=${var2:-"$var1$var1"}
echo -e "\e[01mvar4=\e[00m$var4\r\c"
echo 'VAL4'
40
VAL4=4040
```

explication : "var4=" sera affiché en gras mais sera ensuite écrasée par VAL4, donc il reste juste "=" en gras"

B- Peut-on remplacer \c dans l'avant dernière ligne par autre chose pour obtenir exactement le même résultat? par quoi? (2pt)

par l'option -n de echo

C- Nous avons un dossier "rep" qui contient 14 fichiers nommés fich01, fich02, ... ,fich13, fich14; Nous avons aussi un script shell test.sh executable et contient le code suivant :

```
#!/bin/bash
```

```
echo "$1--$2--$3"
```

- Quel serait le contenu du fichier f après l'exécution de la ligne suivante : (4pt)

```
$ ls rep | grep "1" | tee f | xargs -n 3 test.sh >> f
```

explication: on liste les 14 fichiers, on selectionne ensuite ceux dont le nom contient le chiffre "1", il nous reste 6 noms de fichiers qui seront enregistrés par tee dans le fichier f, ensuite ils seront transmis à xargs qui va les donner 3 par 3 au script test.sh. Ce dernier va les afficher dans le format "\$1--\$2--\$3" et le résultat sera ajouté à la fin du fichier f. Voici le résultat final dans le fichier f :

fich01

fich10

fich11

fich12

fich13

fich14

fich01--fich10--fich11

fich12--fich13--fich14