



Classe:.....

Nom:.....

Groupe: .....

Prénom:.....

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

A- Peut-on accéder au 15<sup>ème</sup> argument d'un script à l'aide des variables positionnelles sous bash? Si oui, comment? Sinon pourquoi? (2pt)

OUI / NON :JUSTIFICATION :

OUI :  `${15}`.....

B-1 Quelle est l'utilité du fichier  `~/.profile` ? (2pt)

initialisations à faire au moment de l'ouverture de la session

B-2 Quelle est la principale différence entre lui et le script  `/etc/profile` ? (2pt)

`.profile` est modifiable par l'utilisateur alors que  `/etc/profile` est modifiable par le superutilisateur

C- Soit un script bash  `controle.sh` qui contient le code suivant :

```
while getopts ":b:set" o; do
    echo -n "$o+"
done
```

C-1 Quelles sont les options traitées par ce script ? (2pt)

`-b` avec valeur.....

`-s` sans valeur.....

`-e` sans valeur.....

`-t` sans valeur.....

C-2 Sachant que le script est exécutable, donnez le résultat affiché à la sortie standard après exécution de cette ligne : (3pt)

```
$ ./controle.sh -se -b
```

`ste+:+`.....



D- Soit les commandes suivantes dans un script bash :

```
set a b c d e f
```

```
t=( $1 $2 $3 $4 $5 )
```

```
echo ${#t[1]}
```

```
echo ${#t[@]}
```

Quel serait le résultat affiché à l'exécution de ces commandes? (3pt)

1.....

5.....

E- Parmi les suivantes, lesquelles ne sont pas des variables spéciales

(2pt)

**\$#**      **\$!**      **\$?**      **\$@**      **\$n**      **\$\***      **\$\$**      **\$%**

F- Soit un script shell exécuté de la manière suivante :

```
$ ./controle.sh > /dev/null 2>&1
```

F-1 qu'est ce que cela signifie ?

(2pt)

que nous redirigeons les deux sorties vers le trou noir

.....

F-2 Donnez une expression équivalente pour faire exactement la même chose :

(2pt)

```
$ ./controle.sh &> /dev/null .....
```

.....