

T.D. 1 : Premières commandes

Exercice 1 On se place dans les conditions suivantes : L'interpreteur est le bourne shell et le prompt de celui-ci \$.

```
$ ls -l a.out
-rwxr-xr-x 1 caron prof 2435 Feb 26 08:02 a.out
$ a.out
sh: a.out: command not found
$
```

1. On suppose que le programme source C correspondant au binaire a.out ne contient pas l'instruction
`printf("sh : a.out : command not found \n");`
Comment expliquer ce comportement et y remédier ?

2. Le fichier source correspondant était en fait :

```
main(){
printf("sh : a.out : command not found");exit(0);}

```

Y a-t-il un moyen de distinguer cette situation de la précédente ?
Si oui lequel ?

3. Modifier le source ci-dessus de telle sorte que si la commande est renommée sous un nom quelconque son exécution donne quel que soit ce nom :

```
$ nom_qcq
sh: nom_qcq : not found

```

Exercice 2 Vous avez tapé la commande suivante et obtenu le résultat correspondant.

```
$ ls *ref*
reference
$
```

Citer trois raisons différentes pour lesquelles la séquence suivante exécutée immédiatement après la précédente, est possible.

```
$ reference
sh: reference: command not found
$
```

Citer d'autres situations pour lesquelles l'exécution de la même commande est susceptible d'échouer également mais avec un message différent.

Exercice 3 On considère la commande suivante et son résultat.

```
$ ls -lia
2234 drwxr-xr-x  4 caron  prof      4096 fév 25 21:21 ./
2542 drwxr-x---  4 caron  prof      4096 fév 25 20:57 ../
2576 drwxr-xr-x  2 caron  prof      4096 fév 25 21:21 t1/
2577 drwxr-xr-x  2 caron  prof      4096 fév 25 21:21 t2/
4578 -rw-r--r--   2 caron  prof        234 fév 25 21:13 td1.aux
4578 -rw-r--r--   2 caron  prof        234 fév 25 21:13 titi
```

Expliciter le nombre de liens pour chacune de ces lignes.

Exercice 4 Supposons maintenant que vous êtes l'utilisateur `lambda`. Vous effectuez la liste des commandes suivantes :

```
$ ls -l
drwxr-xr-x  2 lambda  prof      4096 fév 25 21:43 t1/
$ ls -l t1
-rwx-----  2 root   root      3021 fév 25 21:43 fic
$ cat t1/fic
$ rm t1/fic
$
```

Quels sont les résultats des deux dernières commandes ? Expliciter