

Question I (1.5 points) Que réalise la ligne de commandes :

```
ls -l | grep '\- \- \- ' > resultats
```

Les informations détaillées sur les fichiers/dossiers du répertoire courant (commande `ls -l`) sont filtrées (commande `grep`) pour ne garder que les fichiers/dossiers pour lesquels les utilisateurs "autres" n'ont aucuns droits (le slash permettant de déspecialiser le tiret utilisé par défaut comme un signe d'intervalle).

Le résultat est stocké dans le fichier `resultats`.

Question II (2.5 points) Script bash à écrire :

```
#!/bin/bash                                #!/bin/bash
for f in `ls`                               for i in $*
do                                           do
    for i in $*                             if [ -e $i ]
    do                                       then
        if [[ $f == $i ]]                 echo $i
        then                             fi
            echo $i                       done
        fi
    done
done
```

Question III (6 points) Script système Python à comprendre :

Soit un le fichier nommé `tests` :

```
155.24.157.15 n importe quoi 20160104 n importe quoi
155.24.157.15 n importe quoi 20160104 n importe quoi
127.0.0.1 blabla 20160104
127.0.0.1 blabla 20160105
127.0.0.1 blabla 20160105
```

Exemples d'appel et affichage des résultats :

```
examen.py tests 04 Jan 2016
155.24.157.15 : 2
127.0.0.1 : 1
```

```
examen.py tests 2016 Jan 05
127.0.0.1 : 2
```

```
examen.py
Nombre de paramètres incorrect
```

Expressions régulières :

```
re.search("^\\d\\d$", sys.argv[i])      : vérifie que le deuxième, troisième ou quatrième (vrai) paramètre
                                          contient seulement deux chiffres
re.search("^\\d\\d\\d\\d$", sys.argv[i]) : vérifie que le deuxième, troisième ou quatrième (vrai) paramètre
                                          contient seulement quatre chiffres
re.search("^"+m, sys.argv[i])           : vérifie la présence de la chaîne m dans le deuxième, troisième
                                          ou quatrième (vrai) paramètre
re.search("^((\\d+\\.\\d+\\.\\d+\\.\\d+).+)+ddd, ligne) : extrait quatre nombre séparés par des points
                                          (une adresse IP ?) d'une ligne du fichier
                                          cette adresse IP doit être suivie plus loin sur la ligne par une date
```

Il semblerait que ce script permette une analyse de logs (par exemple d'un serveur web).

(Cette réponse n'était pas demandée).