Déplacement et renommage (l'intitulé était mauvais ;) (2 points)

```
mv Python/p1.sh Shell
mv Shell Bash
```

Recherche (1 point)

```
find . -name "2023*"
ls -R | grep "^2023" (ligne de commande légèrement moins bonne)
```

Expression régulière (deux et seulement deux points) (1 point)

```
^[^\.]*\.[^\.]*\.[^\.]*$
```

Les solutions suivantes ne sont pas vraiment correctes car elles autorisent plus de deux points :

```
[^\.]*\.[^\.]*\.[^\.]*
\.*\.*\.*$
```

Script Bash (6 points)

```
somme=0
for ligne in $(cat $1) # il est possible d'utiliser $@
do
    echo $ligne
    nombre=$(echo $ligne | cut -d, -f1)
    somme=$((somme+nombre))
done
echo $somme
```

J'ai été navré de découvrir la solution tordue (car utilisant une entrée standard) de chatGPT :

```
somme=0
while read ligne
do
    echo $ligne
    nombre=$(echo $ligne | cut -d, -f1)
    somme=$((somme+nombre))
done < $1
echo $somme
```

Je n'ai accordé des points que si elle était correctement programmée...

Fonction palindrome() python (3 points)

Une solution parmi beaucoup d'autres :

```
def palindrome(chaine) :
    for pos in range(len(chaine)) :
        if chaine[pos] != chaine[len(chaine)-pos-1] :
            return False
    return True
```

Une autre solution cryptique de chatGPT que j'ai oublié de désactiver :(:

```
def palindrome(chaine) :
    return chaine == chaine[::-1]
```

Script Python (7 points)

```
with open("systemes.txt", "r") as fd :
    dico = {}
    for ligne in fd :
        ordisSystemes = ligne[:-1].split(":")[1].split(",")
        for ordiSysteme in ordisSystemes :
            ordi = ordiSysteme.split("=")[0]
            système = ordiSysteme.split("=")[1]
            if système in dico :
                dico[système].append(ordi)
            else :
                dico[système] = [ordi]
    nom = input("Système :")
    print(dico[nom]) # et non dico[système] comme 40 d'entre vous ont écrit !
```