Script Shell :
Correction TP2

Exercice 1 : Droits d'accès

1. Essayer de supprimer ou de modifier le fichier /etc/passwd. Que se passe-t-il? Expliquer la situation à l'aide de la commande ls -l.
   Il est impossible de supprimer le fichier /etc/passwd. Seul le propriétaire (root) du fichier peut le supprimer.
   ls -l /etc/passwd
2. A l'aide de la commande id, vérifier votre identité et le(s) groupe(s) au(x)quel(s) vous appartenez.
   id
3. Créer un petit fichier texte nommé "Lecture" (de contenu quelconque), qui soit lisible par tout le monde, mais non modifiable (même pas par vous).
   touch Lecture
   chmod 444 Lecture âpres essayer avec gedit pour modifier lecture
4. Créer un répertoire nommé "Secret", dont le contenu est visible uniquement par vous même.
   mkdir Secret
   chmod 700 Secret
   Les fichiers placés dans ce répertoire sont-ils lisibles par d'autres membres de votre groupe?

Exercice 2 : Redirections, méta-caractères :

Le répertoire /usr/include contient les fichiers d'entête standards en langage C (stdlib.h, ...).

1. Créer un répertoire nommé inc dans votre répertoire de connexion (HOME).
   En utilisant une seule commande, y copier les fichiers du répertoire /usr/include dont le nom commence par std.
   mkdir inc
   cp /usr/include/std* $tp3 inc

cp /usr/include/std* $tp3/inc

dans repertoire root creer un fichier test.sh

cp *.sh $TP inc (TP un repertoire existe deja)

2. Afficher la liste des fichiers de /usr/include dont le nom commence par a, b ou c.

   find /usr/include/ -name [abc]*

3. Modifier la commande de la question précédente pour qu'au lieu d'afficher le résultat, celui-ci soit placé dans un fichier nommé "Abc.list" de votre répertoire de connexion.

   find /usr/include/ -name [abc]* > Abc.list

4. Afficher le contenu de ce fichier en utilisant la commande cat.

   Copier avec cat son contenu dans un nouveau fichier nommé "Copie".

   cat Abc.list > copie

   wc Abc.list

5. Toujours avec cat, créer un nouveau fichier nommé "Double" formé par la mise bout à bout (concaténation) des fichiers "Abc.list" et "Copie".

   Vérifier que le nombre de lignes a bien doublé à l'aide de la commande wc.

   cat Abc.list copie >double

   wc double

6. Créer un fichier nommé "Temp" contenant une ligne de texte.

   touch Temp

7. Avec cat, ajouter la ligne "The end" à la fin du fichier "Temp".

   cat >> temp

   The end

   CTRL d (au clavier)

8. En une seule ligne de commande, faire afficher le nombre de fichiers de /usr/include dont le nom contient la lettre t.

   ls /usr/include/[t]* | wc

   dans un répertoire TP : ls [d]* | wc

Exercice 3 : FIND

1. Afficher la liste des fichiers .h situés sous le répertoire /usr/include.

   find /etc -name "*.conf"

2. Afficher la liste des fichiers plus vieux que 3 jours situés sous votre répertoire de connexion.

   find /etc -ctime +3

Exercice 4 : HEAD, TAIL, TUBES

1. Afficher les 5 premières, puis les 5 dernières lignes du fichier /etc/passwd.
   head -5 /etc/passwd
   tail -5 /etc/passwd
2. Afficher la 7ième ligne de ce fichier (et elle seule), en une seule ligne de commande.
   head -n7 /etc/passwd | tail -n1

Exercice 5 : TR, MORE, TUBES

1. Afficher le fichier /etc/passwd en remplaçant les caractères / par des X.
   Dans le répertoire TP
   Cat Temp | tr / X
2. Obtenir le résultat précédent page par page.
3. **Cat Temp | tr / X | more**

Exercice 6 : Flux d'E/S standards :

Étudier la documentation de la commande tee.

1. Écrire une commande qui affiche "Bonjour" à l'écran et en même temps crée un fichier nommé "Salutation.txt" dont le contenu est le même message.
   echo "Bonjour" |tee salutation.txt
2. Ecrire un script qui lit une ligne de caractères sur son entrée standard et l'écrit sur sa sortie, en passant tous les caractères en majuscules.
   #! /bin/bash
   read a
   echo $a|tr "[a-z]" "[A-Z]"

Exercice 7 : GREP, CUT, UNIQ, SORT ET TUBES :

On dispose d'un fichier texte telephone.txt contenant un petit carnet d'adresses.
Chaque ligne est de la forme "nom prenom numerotelephone". Les champs sont séparés par des tabulations.
Répondre aux questions suivantes en utilisant à chaque fois une ligne de commande shell:

- Afficher le carnet d'adresse trié par ordre alphabétique de noms.
  sort telephone.txt

Cette commande utilise la commande sort pour trier les lignes du fichier texte "telephone.txt" par ordre alphabétique des noms.

- Afficher le nombre de personnes dans le répertoire. **wc -l telephone.txt**
- **La commande wc -l compte le nombre de lignes dans le fichier "telephone.txt", donnant ainsi le nombre de personnes répertoriées.**
- Afficher toutes les lignes concernant les “Dupond”. **grep Dupond telephone.txt**

  La commande grep permet de rechercher toutes les lignes contenant le mot "Dupond" dans le fichier "telephone.txt".
- Afficher toutes les lignes ne concernant pas les “Dupond”.

  cat telephone.txt | grep -v "Dupond"

  La commande grep -v exclut les lignes contenant le mot "Dupond", affichant ainsi toutes les lignes ne concernant pas les "Dupond".
- Afficher le numéro de téléphone (sans le nom) du premier “Dupond” apparaissant dans le répertoire.

  `grep Dupond telephone.txt|uniq -f1|cut -d\; -f3`

  Cette commande utilise grep pour rechercher les lignes contenant "Dupond", uniq -f 1 pour obtenir uniquement la première occurrence, et cut pour extraire le numéro de téléphone (champ 3, séparé par une tabulation).
- Afficher le numéro de téléphone (sans le nom) du premier “Dupond” dans l’ordre alphabétique (ordre basé sur les prénoms)

  sort telephone.txt|grep Dupond |uniq -f 1 | cut -d\; -f 3

  Cette commande utilise sort pour trier les lignes par ordre alphabétique basé sur les prénoms, puis utilise grep, uniq -f 1, et cut de la même manière que la commande précédente pour extraire le numéro de téléphone du premier "Dupond" dans cet ordre.

Exercice 8 : Exo sur les utilitaires, sort (entre autres):

On dispose d'un fichier Matchs.txt contenant le calendrier du *premier tour* des matchs de championnat d’Europe de Football. Le calendrier est constitué de lignes, relative chacune à un match comprenant des informations séparées par une virgule.

Une structure de ligne est la suivante: Pays1,Pays2, Jour, Date, Heure, Stade, Ville .

Exemple de ligne:

France, Roumanie, Vendredi, 10/01/2016,21h, Stade-de-France, Paris

Représente les informations sur le match (France, Roumanie) se déroulant le vendredi 10 juin à partir de 21h au stade de France.

Ecrire un script shell bash permettant de réaliser les opérations suivantes, en affichant chaque fois le numéro de l'opération :

1. Afficher le nombre des matchs du calendrier commençant à 18h
2. Afficher les informations sur les matchs que joue La France.
3. Afficher les informations sur les matchs qui ne se déroulent pas à Paris
4. Afficher les informations sur les matchs qui se déroulent à Lille, Paris, ou Nice.
5. Afficher les informations sur les matchs ayant lieu un Samedi, un Vendredi, ou un Dimanche
6. Afficher les matchs joués par la France à Paris.
7. Trier par ordre alphabétique inverse la liste selon les noms des pays (Pays1, Pays2) participants aux matchs n'ayant pas lieu à Lille.
8. Trier la liste des matchs n'ayant pas lieu à Lille ou à Bordeaux.
9. Afficher le nombre des matchs du calendrier commençant à 21h

```
# Prt_II_B _1.              Afficher le nombre des matchs du calendrier  commençant à 18h
#= = = =
grep 18h Matchs.txt | wc  -l
#= = = =
# Prt_II_B _2.              Afficher  les informations sur les  matchs que joue  La France.
#= = = =
grep -i FRANCE Matchs.txt | grep -v Irlande
#= = = =
Prt_II_B _3.              Afficher les informations sur les matchs qui ne  se déroulent pas à Paris
grep -iv Paris MATCHS.TXT
#= = = =
Prt_II_B _4.              Afficher les informations sur les matchs  qui se déroulent à Lille, Paris, ou Nice.
#= = = =
# le prog bash réalisant cette requête est le suivant:
echo ' '> FTMP
rm FTMP
grep -i LiLLe MATCHS.TXT > FTMP
grep -i Paris MATCHS.TXT >> FTMP
grep -i Nice MATCHS.TXT >> FTMP
more FTMP
```

Ou
grep -E ‘Lille|Paris|Nice’ MATCHS.TXT
#= = = =
Prt_II_B _5. Afficher les informations sur les matchs ayant lieu un Samedi, un Vendredi, ou un Dimanche
#= = = =
le prog bash réalisant cette requête est le suivant:
echo ' '> FTMP
rm FTMP
grep -i Samedi MATCHS.TXT > FTMP
grep -i Vendredi MATCHS.TXT >> FTMP
grep -i Dimanche MATCHS.TXT >> FTMP
more FTMP
#= = = =
Prt_II_B _6. Afficher les matchs joués par la France à Paris.
#= = = =
grep -i France Matchs.txt | grep -i Paris
#= = = =
Prt_II_B _7. Trier par ordre alphabétique inverse la liste selon les noms des pays (Pays1, Pays2) participants aux matchs n'ayant pas lieu à Lille.
#= = = =

grep -v Lille MATCHS.TXT | sort -r

#= = = =
Prt_II_B _8. Trier la liste des matchs n'ayant pas lieu à Lille ou à Bordeaux.
#= = = =
grep -iv Lille MATCHS.TXT | grep -iv Bordeaux | sort
#= = = =