1/ Créez et testez les alias suivants (depuis votre répertoire personnel):

* alias nom=’commande’

2/ Le fichier caché .bashrc est un fichier qui est automatiquement exécuté à chaque démarrage du shell (et donc d’un terminal). Etudiez ce fichier

* less .bashrc
* find -name .bashrc
* ls -al

3/ Ajoutez les alias ci-dessus qui ne sont pas encore présents. Essayez les nouveaux alias. Attention, pour que les nouveaux alias soient connus, il faut redémarrer le terminal.

alias h=’history’

* h
* alias li=’ls -i’
* li
* alias ll=’ls -l’
* ll
* alias la=’ls -a’
* la

4/ Écrire une commande (ou un ensemble de commandes) qui affiche l’occupation disque du répertoire /home. Filtrez et/ou affichez la ligne qui concerne /home.

* df /home

5/ Créez un fichier fruits contenant exactement :

* touch fruits
* nano fruits
  + pomme:2:rouge:1,35:nzelande
  + cerise:6:rouge:3,25:france
  + pomme:3:jaune:2,25:bretagne
  + peche:4:jaune:1,70:espagne
  + peche:5:blanche:1,60:france
  + banane:9:jaune:1,15:guadeloupe
  + pomme:4:jaune:1,10:france
* Ctrl+X
* O
* Entrée
* a°) cut -d: -f 1,7 fruits
* b°) cut -d: -f 1,3 fruits
* c°) cut -d: -f 1,4,5 fruits

6/ Utiliser la commande head pour n’afficher que les 3 premières lignes du fichier fruits, puis utiliser la commande tail pour n’afficher que les 2 dernières lignes. Utilisez ces deux commandes pour n’afficher que les lignes 3 et 4 du fichier des fruits.

* head fruits -n3
* tail fruits -n2
* head fruits -n4 | tail -n2

7/ Toujours en utilisant la commande cut, donnez la liste des GIDs qui sont présents dans /etc/passwd.

* cut -d: -f3 /etc/passwd

8/ Analysez le type des fichiers présents dans /etc (Utilisez ll ou regardez la documentation de la commande file)

* ll /etc/
* La plupart des fichiers présents dans /etc, sont des fichiers de configuration.

9/ Que fait la commande find . -iname \*.txt -exec wc -l {} \;

* La commande find …, affiche tous les fichiers .txt et nous montre uniquement les fichiers que l’on peut ouvrir.

10/ Après avoir créé un répertoire Copie/ dans votre répertoire concernant le Système, testez le résultat de la commande suivante : find -iname \*.txt -exec echo cp {} Copie \; Que se passera-t-il si vous supprimez echo de la commande ?

* Si l’on supprime echo, il n’y a plus d’affichage de ce que fait la commande.

11/ En utilisant find, cherchez dans /etc les fichiers dont le nom se termine par .conf. Pour chaque fichier trouvé, affichez son contenu (utilisez exec).

* find \*.conf
* Une fois les fichiers ouverts, nous pouvons constater que tous les fichiers sont des fichiers de configuration.

12/ Faites une archive de votre répertoire personnel et compressez-là.

* tar -cvf archive.tar.gz ./\*

13/ Décompresser et extrayez les fichiers de votre archive compressée.

* Tous nous fichiers sont dans le fichier zipée

14/ Que font les commandes suivantes ?   
mkdir redirection -> Créée un répertoire “redirection”  
echo “Bonjour” -> Affiche “Bonjour”  
echo “Bonjour”> redirection/bonjour.out => Créée un fichier bonjour.out dans le répertoire redirection et écrit Bonjour dedans.

echo “Salut” > redirection/bonjour.out => Ecrit Salut dans le fichier bonjour.out

echo “Bonjour” » redirection/bonjour.out => Ecrit bonjour dans le fichier bonjour.out

15/ Dans un répertoire de tests, créez les fichiers suivants : prog.c prog.o projet.c projet.o projet.out presentation scene Donner les commandes utilisant les plus courtes possibles pour effectuer les opérations demandées ci-dessous.

1. Afficher prog.c et prog.o
   * ls tests/prog.c tests/prog.o
2. Afficher prog.c et projet.c
   * ls tests/projet.c tests.prog.c
3. Afficher projet.o et projet.out (mais pas projet.c ni les autres)
4. Afficher projet.c, projet.o, projet.out et présentation
5. Afficher tous les fichiers sauf présentation
6. Afficher projet.c et projet.o mais pas projet.out
7. Afficher presentation et scène mais aucun autre
8. Afficher scène seulement
9. Afficher projet.out et présentation

**Shell et variables**

16/ Exécutez et expliquez les commandes suivantes :

*message1=Bonjour => enregistre le mot “Bonjour” dans la variable “message1”*

*echo $message1 = >renvoie bonjour*

*message2=message1 => enregistre le mot “message1” dans la variable “message2”*

*echo $message2 => renvoie message1*

*unset message2 => clear la variable message2*

*echo $message2 => renvoie rien*

*message3= => enregistre rien dans la variable “message 3”*

*echo $message3 => renvoie rien*

*message4 = Bonjour => La commande “message4” n’a pas été trouvée*

*echo $message4 => renvoie rien*

*bash => efface toute les mémoires des variables.*

*echo $message1 => renvoie rien*

17/ Portée d’une variable

— Créez une variable varp=exemple

— Affichez la valeur de varp : echo $varp

— Lancez un deuxième shell (en premier plan). Pour tester votre environnement, contrôlez la disponibilité de la variable varp dans le nouveau shell : echo $varp

— Quittez le deuxième shell (revenez au premier). Contrôlez la disponibilité de la variable varp On va étudier l’export des variables un peu plus loin. :

18/ Modification d’une variable

— Créez maintenant la variable varx en lisant son contenu du clavier (commande read)

— Affichez la valeur de varx => echo $vrax

— Modifiez la valeur de varx en dupliquant son contenu => vrax=$vrax$vrax

— Affichez la nouvelle valeur de varx => echo $vrax

**Variables d’environnement**

19/ Affichez les variables d’environnement existantes avec la commande printenv (ou encore set | more si besoin). Analysez cette liste et essayez de deviner à quoi servent ces variables (suivez vos bonnes intuitions).

* printenv

20/ Affichez le contenu de la variable PATH. Vous pouvez y observer une liste. Quel est le séparateur des éléments ? Quel est le rôle de cette variable ?

* On y observe une liste de chemin.
* Les séparateur sont les ‘:’.
* La variable PATH, est une variable utiliser pour des valeur par défault.

21/ Créez un répertoire bin dans votre répertoire personnel (HOME) et copiez-y quelques fichiers exécutables de /bin (par exemple : echo, ls, pwd,...) en changeant leur nom, par exemple : contre monecho, monls, etc. Vérifiez les droits avec ls -l bin/\*

* Le super user à tous les droit.
* Le user et le groupe ont les droit rx

22/ Essayez d’exécuter un des programmes déposés dans le répertoire ~/bin en appelant directement le nom court du programme depuis votre répertoire personnel (par exemple monpwd si c’est son nouveau nom dans ~/bin). Allez dans le répertoire ~/bin et appelez la commande en tapant son nom (monpwd). Appelez maintenant la commande par ./monpwd. Que peut-on voir et déduire ?

* La commande ./ permet d’exécuter des fichiers qui ont les permissions d’exécutables.

23/ Ajoutez ce répertoire à la liste dans PATH. Essayez d’exécuter un des programmes déposés dans le répertoire ~/bin en appelant directement le nom du programme depuis un autre répertoire.

* path=/bin/pydoc2
* ./$path

24/ Observez le contenu des variables HOME, PATH, OLDPWD, PS1 et PS2. Quel est le rôle de ces variables ?

* /home/depinfo1
* /home/depinfo1/bin:/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin
* /bin
* \[\e]0;\u@\h: \w\a\]${debian\_chroot:+($debian\_chroot)}\[\033[01;32m\]\u@\h\[\033[00m\]:\[\033[01;34m\]\w\[\033[00m\]\$
* >

25/ Changez les messages PS1 et PS2.

26/ Créez une variable d’environnement PROMO avec le contenu ’1/2’. Vérifiez PROMO dans des shells fils.

PROMO='1/2'

echo $PROMO

1/2

**Le shell est interactif**

27/ Lancez un éditeur gedit en arrière plan (gedit&). Comment l’utiliser ? Comment le quitter ?

* Un éditeur gedit permet d’écrire du texte dans un fichier .txt. On l’utilise comme un Word ou un block-note. On le quitte avec CTRL+C ou la croix en haut a gauche.

28/ Lancez un nouveau shell en arrière-plan (bash&). Comment l’utiliser ? Comment le quitter ?

* On l’utilise avec un shell normal. On le quitte avec CTRL+D

Quelle est la différence entre les deux types de programme (éditeur avec interface graphique et shell) ?

* La différence entre éditeur avec interface graphique est shell, n’est pas si grande, juste l’affichage et le fait que avec le shell l’on peut accedez a distance avec un SSH.

**Substitution des commandes et des variables**

29/ Dans cet exercice, faites attention à la différence entre les délimiteurs ’ ’ puis ‘ ‘ puis " ". Exécutez et expliquez ce que font les commandes suivantes :

echo ’ls -l’

* La commande permet d’afficher les mots ls -~~l~~ et quand celui-ci s’arrête.

echo ‘ls -l‘

* Affiche juste les mots / phrase entre ‘’

echo "je m’appelle $USER"

* Affiche ce qu’il y a en parametre en plus modifier les variable avec leur contenue.

ls ‘ls‘

* Cherche un dossier/fichier ‘ls’ dans le dossier courant.

Expliquer les effets des délimiteurs ’ ’ puis ‘ ‘ puis " ".

30/ Positionnez-vous dans un répertoire quelconque dans le système. Affichez la première et la deuxième partie du nom absolu du répertoire ou vous êtes.

* pwd | cut -d/ -f2,3

31/ On propose de faire un petit dialogue avec le shell. Le scénario envisagé est le suivant :

machine > Bonjour, qui es-tu ?

utilisateur > Mathilde

machine > Bonjour Mathilde

echo "machine> Bonjour, qui est-tu ?"

echo "utilisateur > Mathilde"

echo "machine > Bonjour $persona"

32/ Trouvez les lignes contenant bla.

grep bla TEXTE.txt

33/ Trouvez les mots se terminant par ques.

cat TEXTE.txt | grep "ques\>"

34/ trouver des lignes vides.

cat TEXTE.txt | grep -vn .

35/ Trouvez les lignes commençant par une voyelle.

cat TEXTE.txt | grep ^[aeiouyAEIOUY]

36/ Trouvez les lignes commençant par une voyelle ou par un chiffre.

Cat TEXTE.txt | grep ^[aeiouyAEIOUY] | grep ^[0-9]

37/ Trouvez les lignes commençant par une voyelle et se terminant par le caractère ?

Cat TEXTE.txt | grep ^[aeiouyAEIOUY] | grep ?$

38/ Trouvez les lignes ne commençant pas par une voyelle.

cat TEXTE.txt | grep ^[^aeiouyAEIOUY]

39/ Recherchez tous les fichiers du répertoire /etc qui appartiennent à root (leur ligne dans la liste étendue des fichiers contient la chaîne de caractères ” root ”.

cd /etc

ls -ali | grep root

40/ Recherchez les fichiers du répertoire /etc qui ne peuvent pas être exécutés par les ”autres” utilisateurs.

ls -ali | cut -d' ' -f2 | grep [-]$

41/ Recherchez les fichiers du répertoire /etc qui ne peuvent pas être exécute par personnes.

ls -ali | cut -d' ' -f2 | grep -v [x]