

INF3135 – Démo 1

© Emmanuel Chieze

Programmation en C : tables de multiplication (utilisation de boucles for, sorties formatées)

Écrire un programme qui affiche les tables de multiplication comme suit :

```
* | 1 2 3 4 5 6 7 8 9 10
-----
1 | 1 2 3 4 5 6 7 8 9 10
2 | 2 4 6 8 10 12 14 16 18 20
...
10 | 10 20 30 40 50 60 70 80 90 100
```

Vous utiliserez le format `%4d` pour afficher un entier sur 4 caractères, cadré à droite, avec l'espace comme caractère de remplissage.

UNIX: commandes de base

1.1. Accès au réseau

Au laboratoire, il suffit de taper votre nom d'utilisateur et votre mot de passe à l'invite de Linux.

Si vous désirez accéder au réseau de l'extérieur, le moyen le plus simple est d'utiliser SSH avec l'adresse `rayon1.labunix.uqam.ca` ou `zeta.labunix.uqam.ca`, selon le serveur auquel vous désirez vous connecter.

1.2. Pour quitter le réseau

Au laboratoire, utilisez l'option "Close session" du menu principal (le bouton en bas à gauche de l'écran).

Lorsque vous travaillez de l'extérieur avec SSH, tapez la commande "exit" à l'invite.

1.3. La commande `passwd`

Permet de modifier votre mot de passe. Par exemple:

```
% passwd
passwd: Changing password for e12345
Enter login(NIS) password:
```

New password:
Re-enter new password:

1.4. La commande ls

Donne une liste des fichiers du répertoire courant. Par exemple:

```
% ls
#essai.pl#  nsmail/
On peut aussi utiliser l'option -l pour voir les détails sur chaque fichier:
% ls -l
total 4
-rw-----  1 e12345  pj110570      440 sep 15 18:53 #essai.pl#
drwx-----  2 e12345  pj110570      512 sep 15 17:33 nsmail/
```

ATTENTION : UNIX est sensible à la casse (case-sensitive)

1.5. La commande cd

Change le répertoire courant. On peut lui donner un chemin absolu ou relatif au répertoire courant. Par exemple (relatif):

```
arabica(p35470) [22] /usagers/p35470
% cd Tp1
arabica(p35470) [23] /usagers/p35470/Tp1
```

Relatif; répertoire spécial "." (parent):

```
arabica(p35470) [23] /usagers/p35470/Tp1
% cd ..
```

```
arabica(p35470) [24] /usagers/p35470
```

Chemin absolu (commence par "/", la racine):

```
arabica(p35470) [24] /usagers/p35470
% cd /usr/bin
```

```
arabica(p35470) [25] /usr/bin
```

cd utilisé seul; ramène à votre répertoire personnel (HOME):

```
arabica(p35470) [25] /usr/bin
% cd
arabica(p35470) [26] /usagers/p35470
```

1.6. La commande cp

Copie un fichier. On écrit "cp <fic_source> <fic_destination>". Par exemple:

```
% cp tp1.pl copie_de_sauvegarde_tp1.pl
```

1.7. La commande mv

Déplace ou renomme un fichier. On écrit "mv <fic_source> <fic_destination>". Par exemple:

```
% mv essai3.pl tp1.pl
```

1.8. La commande rm

Efface un fichier. Par exemple:

```
% rm essai3.pl
```

1.9. La commande mkdir

Crée un répertoire. Par exemple:

```
% mkdir Tp1
```

1.10. La commande rmdir

Détruit un répertoire. Par exemple:

```
% rmdir Temporaire
```

1.11. La commande more

Permet de faire afficher un fichier ou la sortie d'une autre commande écran par écran. Par exemple:

```
% more tp1.pl
```

Pour capturer la sortie d'une autre commande, on tape cette commande suivie d'une barre verticale et de "more". Par exemple:

```
% ls -l | more
```

Pour passer à la page suivante, on utilise la barre d'espace. Pour quitter, il suffit de taper "q".

1.12. La commande script

Permet de capturer la sortie à l'écran et de la sauvegarder dans un fichier. Pour démarrer, on tape "script <nom_fic>". Par exemple:

```
% script capture_tp1
```

Script démarré - résultat dans le fichier capture_tp1.

Ensuite, on peut travailler normalement et tous les résultats seront sauvegardés dans le fichier. Pour terminer l'enregistrement, il faut taper "exit" ou appuyer sur Ctrl-D. Par exemple:

```
% exit
```

```
exit
```

Script terminé - résultat dans le fichier capture_tp1.

1.13. La commande grep

Fait une recherche dans un fichier et renvoie les lignes trouvées. On écrit "grep <expression> <fichier>". Par exemple:

```
% grep fruit test.pl
```

```
aliment(pomme, fruit).
```

```
aliment(banane, fruit).
```

```
aliment(tomate, fruit).
```

On peut utiliser l'option -i pour que grep ne tienne pas compte de la casse:

```
% grep -i FRUIT test.pl
```

```
aliment(pomme, fruit).
```

```
aliment(banane, fruit).
```

```
aliment(tomate, fruit).
```

On peut aussi utiliser l'option -n pour que grep indique les numéros de lignes:

```
% grep -n fruit test.pl
5:aliment(pomme, fruit).
6:aliment(banane, fruit).
9:aliment(tomate, fruit).
```

1.14. La commande man

Donne le manuel d'instructions pour une commande UNIX. Par exemple:

```
% man rm
Mise en page en cours, veuillez patienter... Ok

User Commands                                                                    rm(1)

NAME
    rm, rmdir - remove directory entries

SYNOPSIS
    /usr/bin/rm  [ -f ]  [ -i ] file ...

    /usr/bin/rm -rR  [ -f ]  [ -i ] dirname ... [ file ... ]

    /usr/xpg4/bin/rm  [ -fiRr ] file ...

    /usr/bin/rmdir  [ -ps ] dirname ...

DESCRIPTION
    /usr/bin/rm /usr/xpg4/bin/rm
    The rm utility removes the directory entry specified by
    each file argument. If a file has no write permission and
    the standard input is a terminal, the full set of permis-
    sions (in octal) for the file are printed followed by a
    question mark. This is a prompt for confirmation. If the

etc.
```

Comme avec la commande more, on utilise la barre d'espace pour passer à la page suivante et "q" pour quitter.

1.15. Gestion des droits d'accès et commande chmod

Notions d'utilisateur, de groupe et d'autre

Droits d'accès des fichiers

Droits d'accès des répertoires

Commande chmod