

---

# TP 3 VARI 1

Informations techniques PC Suse :

- (a) Pour démarrer une session : utilisateur **licencep** et mot de passe **7002n\***. Vous trouverez :
  - un gestionnaire de fichiers en haut à gauche placé dans le dossier personnel HOME
  - une icône lézard  en haut à droite pour accéder au menu.
- (b) Pour démarrer *Processing* : clic sur l'icône lézard en haut à droite → Développement → Processing.
- (c) Pour démarrer une *console* : clic sur l'icône lézard en haut à droite → Terminal → Konsole.
- (d) Pour ouvrir un gestionnaire/navigateur de fichiers : clic sur l'icône lézard → Utilitaires → Dolphin.
- (e) Pour lancer une commande : Alt + Space
- (f) Pour modifier un fichier, clic droit sur le fichier → Ouvrir avec KWrite (ou autre éditeur de votre choix).

---

## 1 Commandes Linux : systèmes et réseaux

Démarrer une console/terminal en suivant les instructions au point (c) ci-dessus.

**Exercice 1** Taper la commande suivante dans le terminal *Shell*, en faisant attention aux espaces. Observez la version du noyau, le type de processeur et l'OS générique (Gnu/Linux).

```
uname -rop
```

**Exercice 2** Taper les commandes suivantes une par une. On observe que la première commande indique l'OS plus précisément : SUSE Linux. Dérouler le résultat de la deuxième commande pour trouver combien de coeurs possède le processeur (CPU).

```
cat /proc/version
cat /proc/cpuinfo
```

**Exercice 5** Donner une commande `nslookup` pour trouver l'adresse IP de `www.google.fr`.

**Exercice 6** Quel est le résultat de la commande `ping -b 163.173.231.255`? Quelle est l'adresse de diffusion/broadcast de votre machine?

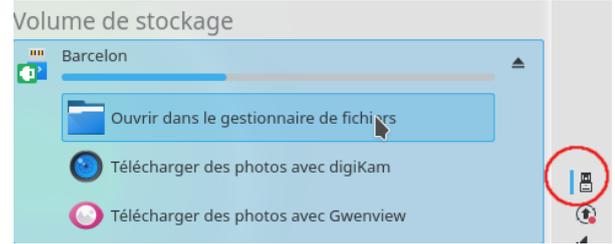


**Rappel :** pour **rechercher** dans le manuel une option d'une commande (ex., l'option `-b` de la commande `ping`), taper "`man NOM.COMMANDE`" (ex., `man ping`) et taper `/`, suivi de la chaîne de caractères recherchée (ex., `-b`) et d'*Entrée*. Ensuite, taper `n` pour rechercher l'apparition suivante du mot. Pour quitter le manuel, utiliser `q`.

**Exercice 7** Travailler en binôme sur deux machines *A* et *B*. La machine *A* lance `netcat -l 10000` pour ouvrir un serveur TCP qui écoute le port 10000 (option `-l`=listen). La machine *B* lance `netcat IPA 10000`, où *IP<sub>A</sub>* est l'adresse IP de *A*. Vous obtenez une connexion TCP et vous pouvez communiquer comme dans un programme de messagerie instantanée.

**Exercice 8** Lancer un `ping` sur la machine `samar31` (ex., `ping -c 3 samar31`) et noter son adresse IP. Se connecter à `samar31` (ou `samar32`, `samar33`, etc) via `ssh licencep@samar31`. Vous avez accès à `samar31`, soyez sages svp! Taper `nmmap samar31` ou `nmmap samar32` pour voir les ports TCP ouverts et finir avec `exit`.

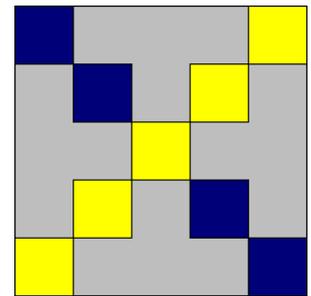
**Exercice 9** Brancher une clé USB, remarquer le logo “clé USB” en bas à droite (voir cercle rouge, figure à droite), cliquer sur “Ouvrir dans le gestionnaire de fichiers”. Finalement, taper la commande `mount`. Les dernières lignes indiquent un texte comme `/dev/sdb1 on /run/media/USBDISK1`. C’est le dossier où la clé a été montée. Se placer dans ce dossier (commande `cd`). Taper une commande `ls` pour lister les fichiers et les dossiers sur la clé.



## 2 Processing

**Exercice 1** Soit le code ci-dessous. Ajouter la deuxième diagonale composée de petits carrés jaunes pour faire le dessin en bas à droite. Pour obtenir la couleur jaune, on utilise une combinaison de rouge et vert !

```
1 size (250,250);
2 fill (190,190,190); //mélange rouge, vert, bleu=>gris
3 rect (0,0,250,250);
4 fill (0,0,120); //un peu de bleu 120
5 rect (0,0,50,50);
6 rect (50,50,50,50);
7 rect (100,100,50,50);
8 rect (150,150,50,50);
9 rect (200,200,50,50);
```



**Exercice 2** Soit le code ci-après, qui n’utilise pas de variable entière, mais uniquement des `string`, c.t.d., chaînes de caractères.

- Corriger l’erreur à la première ligne.
- Pour quelles valeurs de la variable `monAge` (vue comme une chaîne de caractères!), le programme afficherait “Je suis jeune” ou “Je suis vieux”? Info : `monAge.length()` est la longueur de la chaîne.
- Doubler la taille du texte aux lignes 8 ou 10 sans changer la taille du texte à la ligne 5

```
1 size (400,100)
2 String monNom="Pierre"; //declaration variable String (chaîne caract.)
3 String monAge="100"; //pas de variable int/float
4 textSize (25);
5 fill (0,0,0);
6 text ("Je_suis_" + monNom + "_et_j'ai_" + monAge + "_ans", 0,20);
7 if (monAge.length () < 2)
8   text ("Je_suis_jeune", 0,40);
9 if (monAge.length () > 2)
10  text ("Je_suis_vieux", 0,40);
```

**Exercice 3** Déclarer une variable entière  $x$  (de type `int`) et une variable réelle  $y$  (de type `float`). Initialiser  $x = 2$  et  $y = 0.5$ . On veut stocker dans une variable  $z$  la somme  $z = x + y$ . Déclarer la variable  $z$  (quel type choisir?), faire l’affectation  $z = x + y$  et finalement afficher  $z$  à l’aide de la fonction `println(...)`.

**Exercice 4** Déclarer une variable  $x$  de type `String` (chaîne de caractères) et une variable entière  $y$ . Initialiser  $x = \text{"salut"}$  et  $y = 9$ . Quel est le type de la variable  $z$  qui permet d’exécuter l’affectation suivante?

```
z=x+y
```

Déclarer la variable  $z$ , faire l’affectation ci-dessus et afficher  $z$ . Par la suite appeler la fonction `println(z+z+z);`. Vous pouvez deviner son résultat avant de l’exécuter ?

**Exercice 5** Soit les variables `exam`, `tp1`, `tp2`, `tp3` et `tp4` initialisées au début du programme pour indiquer une note d’examen et resp. 4 notes de TP. Une note d’examen inférieure à 7 est éliminatoire. La plus petite note des quatre notes de TP est ignorée. La note finale est calculée ainsi : l’examen compte pour 60% et la moyenne des trois notes de TP restantes pour 40%. Exemple : si  $exam = 10$ ,  $tp1 = 5$ ,  $tp2 = tp3 = tp4 = 12$ , on obtient  $10 \cdot 0.6 + 12 \cdot 0.4 = 10.8$ . Si la note finale est supérieure à 10, afficher “réussi”, sinon afficher “échoué”.