
TP 5 VARI 1

cedric.cnam.fr/~porumbed/vari1/

Informations techniques PC Suse :

- (a) Pour démarrer une session : utilisateur **licencep** et mot de passe **7002n****.
- (b) Pour démarrer *Processing* : clic sur la tête de coméléon en haut à droite → Développement → Processing.
- (c) La page VARI1 : cedric.cnam.fr/~porumbed/vari1/
- (d) Pour ouvrir un gestionnaire/navigateur de fichiers : clic sur la tête de caméléon → Système → Dolphin.

Exercice 1 Télécharger la machine virtuelle **AntixLinux** (avec Processing) à l'adresse suivante : <http://cedric.cnam.fr/~porumbed/vari1/vbox/>

Démarrer cette machine virtuelle à l'aide des instructions fournies à l'adresse ci-dessus. Nous allons faire les exercices suivants sur cette machine virtuelle, car elle nous donne un accès administrateur.

Exercice 2 Démarrer un terminal (CTRL-gauche⊕Alt⊕ t) et placez vous dans le dossier **/usr/** à l'aide de la commande `cd /usr`. Lister (afficher) les fichiers et les dossiers à l'intérieur de **/usr/** à l'aide de la commande `ls`.

Exercice 3 Taper juste `cd` pour se placer dans le dossier personnel. Créer un dossier **vari1** à l'aide de la commande `mkdir vari1`. Finalement, taper `cd vari1` pour se placer dans le dossier que vous venez de créer.

Exercice 4 Taper `echo "Toto"> fichier.txt` pour créer un fichier **fichier.txt** dans lequel vous allez trouver le contenu « Toto ». Taper par la suite la commande `ls` pour vérifier que le fichier est bel et bien à l'intérieur du dossier courant. Taper aussi `cat fichier.txt` pour voir le contenu de ce fichier **fichier.txt**.

Exercice 5 Taper dans un terminal la commande ci-après, pour télécharger le sujet de TP d'aujourd'hui.

```
wget cedric.cnam.fr/~porumbed/vari1/tp5.pdf
Taper ls pour vérifier que vous avez un nouveau fichier
tp5.pdf. Ouvrir ce fichier avec la commande ci-après
(le & sert à lancer la commande en « background »,
c.à.d., en arrière-fond pour ne pas bloquer le terminal) :
evince tp5.pdf &
```

Exercice 6 Taper les commandes suivantes pour installer le logiciel **pdftk** et pour extraire la première page du fichier **tp5.pdf** :

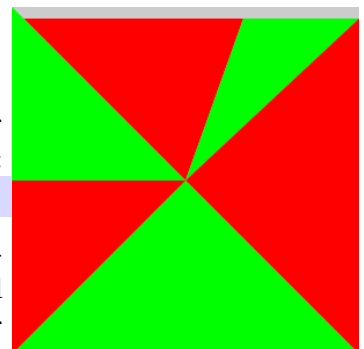
```
sudo apt-get update
sudo apt-get install pdftk
pdftk tp5.pdf cat 1 output page1.pdf
evince page1.pdf
```

Exercice 7 Taper les commandes ci-après pour télécharger deux fichiers pdf :

```
wget cedric.cnam.fr/~porumbed/vari1/totonobel.pdf
wget cedric.cnam.fr/~porumbed/vari1/totosign.pdf
```

Taper `evince totonobel.pdf` pour regarder le premier fichier. On veut signer le fichier **totonobel.pdf** à l'aide de la signature dans **totosign.pdf**. Il suffit de taper : `pdftk totonobel.pdf stamp tosign.pdf output toto.pdf`

Exercice 8 Utiliser six appels `triangle(...)` pour réaliser le dessin à droite, en deux couleurs (rouge et vert). **Indication** : L'appel `triangle(x1, y1, x2, y2, x3, y3)` permet de tracer un triangle déterminé par les points/sommets (x_1, y_1) , (x_2, y_2) , (x_3, y_3) .



Exercice 9 Utiliser deux appels `quad(...)` pour réaliser le dessin à droite, de couleur verte. Indication :

- L'appel `quad(x1, y1, x2, y2, x3, y3, x4, y4)` permet de tracer un quadrilatère déterminé par les points/sommets (x_1, y_1) , (x_2, y_2) , ...



Sortir de la machine virtuelle avant de continuer : (CTRL-droit⊕P)

Exercice 10 Écrire un programme `processing` qui permet d'initialiser une variable `salaire` de type `float` à une valeur au hasard entre 300 euros et 1000000 euros. Afficher un des messages suivants :

- Si le salaire est inférieur ou égal à 10000, afficher « Vous n'êtes pas imposable »
- Si le salaire est supérieur à 10000 mais inférieur ou égal à 100000, afficher « Vous payez un impôt de 10% ».
- Si le salaire est supérieur à 100000, afficher « Vous êtes footballeur? Vous payez un impôt sur la fortune de 30% »

Finalement, afficher le salaire **net**, après la déduction de l'impôt.

Exercice 11 Écrire un programme qui permet de tracer un hexagone de couleur rouge, sans contour; vous pouvez utiliser deux appels à `quad(...)`. À l'intérieur de l'hexagone, tracer un cercle de couleur bleu et écrire «Salut Toto » à l'intérieur du cercle.

Exercice 12 Corriger les erreurs de compilations du programme ci-après et exécuter ce programme.

```
1 int zEntier=0, x = 4;
2 float z = random(3)
3 String str = "x="+x; //str est une chaine de caractères
4 println(str); //on affiche cette chaine qui comporte la valeur de x
5 println(str.length()); //on affiche le nombre de caractères, toujours 3, pourquoi?
6 int zEntier = (int)z //le '(int)' devant 'z' fait une conversion vers int
7 if(zEntier===2)
8     println("z vaut deux");
```

Exercice 13 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 14 Modifier le programme ci-après pour le faire rebondir la balle lorsqu'elle touche le bord à droite. N'hésitez pas à regarder le début de la vidéo 5 pour vous familiariser avec les méthodes `setup()` et `draw()`.

```
1 int x; //variable globale visible dans chaque méthode
2 void setup() {
3     size(600,600);
4     x = 0;
5 }
6 void draw() {
7     background(100,100,100); //effacer la toile
8     x = x + 1;
9     ellipse(x,300,10,10);
10 }
```

Exercice 15 Ouvrir un terminal sur la machine physique : tête lézard en haut à droit → Terminal → Konsole. Taper la commande suivante pour télécharger le fichier `cercles.png` et utiliser le gestionnaire de fichiers, voir point (c) tout au début du document pour l'ouvrir :

```
wget cedric.cnam.fr/~porumbed/vari1/cercles.png
```

Réaliser ce dessin sous `processing`. Vous allez avoir besoin de 5 couleurs olympiques à obtenir avec les appels : `stroke(21,195,245)`, `stroke(0,0,0)`, `stroke(184,20,55)`, `stroke(245,184,8)`, `stroke(49,255,0)`.

Exercice 16 BONUS. Continuer l'exercice 15 précédent pour réaliser le logo olympique disponible à l'adresse : `wget cedric.cnam.fr/~porumbed/vari1/drapeauolympique.png`

Ce dessin a été réalisé avec `Processing` grâce à plusieurs appels `arc(...)`. Certaines sections des cercles en haut doivent effacer les cercles en bas.