
TP 6 VARI 1

Web : 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.
-

1 Le terminal

Exercice 1 Démarrer un terminal : il suffit de taper `CTRL-gauche⊕Alt⊕t`. Par défaut, le terminal est placé dans le dossier personnel. Taper `pwd` pour voir l'emplacement du dossier personnel. Démarrer en même temps un gestionnaire graphique de fichiers comme indiqué au point (d) ci-dessus. Taper la commande `mkdir tp6vari1` dans le terminal pour créer un dossier dans le dossier personnel. Placez vous dans ce dossier à la fois avec le gestionnaire graphique (cliquer sur `tp6vari1`) et avec le terminal (taper `cd tp6vari1`).

Exercice 2 Taper dans le terminal la commande ci-après, pour télécharger le sujet d'aujourd'hui.

```
wget cedric.cnam.fr/~porumbed/vari1/6tp.pdf
```

Exercice 3 Utiliser le gestionnaire graphique de fichiers pour ouvrir le pdf téléchargé plus haut.

Exercice 4 Taper la commande `touch toto1.txt` pour créer un fichier `toto1.txt`. Appuyer sur « flèche haute » pour retrouver la dernière commande. Remplacer «1» avec «2» suivi de «Entrée» pour lancer la commande `touch toto2.txt`. Répéter le processus pour créer 7 fichiers `toto1.txt`, `toto2.txt`, ..., `toto7.txt`.

Exercice 5 Utilisez le gestionnaire graphique et cliquez sur le fichier `toto6.txt` pour l'ouvrir avec un éditeur. Écrire un petit texte dans ce fichier et sauvegardez-le (`CTRL⊕S`). Taper `cat toto6.txt` pour afficher ce texte dans le terminal.

Exercice 6 Taper la commande `man cp` pour lire le manuel de la commande `cp`; une fois la lecture finie, taper `q` pour sortir du manuel. Copier le fichier `toto6.txt` vers `gogo6.txt` à l'aide d'une commande `cp`, où il faut remplacer/remplir les points. Afficher le contenu du fichier `gogo6.txt` et vérifiez que vous trouvez le texte que vous avez tapé dans l'éditeur de l'exercice précédent.

Exercice 7 Taper la commande `ls` pour voir les fichiers dans le dossier courant. Qu'est est le résultat de la commande `ls go*` ?

Exercice 8 Taper la commande `man rm` pour lire le manuel de la commande `rm`; une fois la lecture finie, taper `q` pour sortir du manuel. Taper la bonne commande `rm ...` pour effacer le fichier `gogo6.txt`. Vérifiez avec le gestionnaire graphique de fichiers que ce fichier est bien effacé.

2 Processing

Exercice 9 Soit le code à droite. La méthode `setup()` initialise une toile de taille 600×600 et la Ligne 3 indique que le programme doit appeler `draw()` 3 fois par seconde. Chaque appel à `draw()` permet de tracer un cercle. Modifier ce code pour le faire tracer des cercles de couleurs aléatoires avec un contour très épais de couleur aléatoire aussi.

```
1 void setup() {
2   size(600,600);
3   frameRate(3); //3 par seconde
4 }
5 void draw() {
6   float diametre = random(600);
7   ellipse(300,300,diametre , diametre);
8 }
```

Exercice 10 Initialiser une toile de taille 600×600 et tracer deux cercles de rayon $r = 150$ à des positions aléatoire sur la toile. Utiliser la conditionnelle `if` pour déterminer si les deux cercles se croisent. Deux cercles se croisent si la distance entre leur centres est inférieure à $2r$. Pour calculer la distance entre les deux centres, il faut utiliser le théorème de Pythagore (et la fonction `sqrt(...)` pour déterminer la racine carré). Afficher sur la toile un texte de grande taille qui indique si les deux cercles se croisent ou pas.

Exercice 11 Initialiser une toile de taille 600×600 et tracer un cercle de rayon $r = 300$ et centre $(300, 300)$. Tracer une ligne à une position aléatoire sur la toile. Vérifier si cette ligne reste complètement à l'intérieur du cercle. Cela se passe uniquement si les deux extrémités de la ligne se situent à une distance inférieure à $r = 300$ du centre $(300, 300)$. Afficher un texte sur la toile pour indiquer si cette ligne est bien à l'intérieur du cercle ou pas.

Exercice 12 Quel est le résultat du code ci-après ? Corriger une erreur.

```
for (int i=0; i < 10, i++){
    float x=random 20;
    println ("x="+x);
}
```

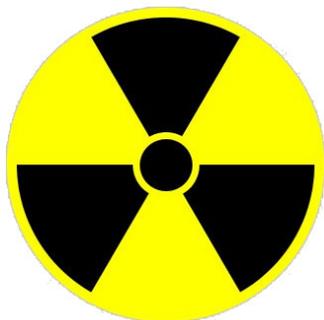
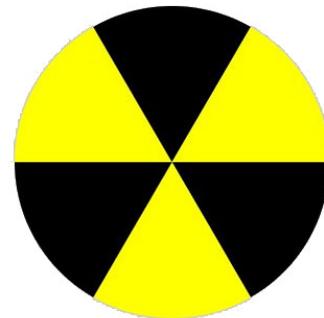
Exercice 13 Écrire un programme qui permet de tracer le dessin à droite une seule fois. Modifier le programme pour afficher le même dessin 15 fois à 15 positions aléatoires sur la toile.



Exercice 14 Soit le code ci-dessous qui permet de tracer l'image à droite.

```
size(700,700);
noStroke();
fill(255,255,0); //faire un grand cercle
ellipse(350,350,300,300); //jaune
fill(0,0,0); //Ajouter trois arcs noir
arc(350,350,300,300,0,0 + PI/3);
arc(350,350,300,300,2*PI/3,2*PI/3 + PI/3);
arc(350,350,300,300,4*PI/3,4*PI/3 + PI/3);
```

Modifier ce programme pour obtenir l'image à droite, le symbole d'un avertissement nucléaire. Vous allez avoir besoin de tracer (au centre) un petit cercle avec un contour de couleur jaune ; pour sélectionner la couleur du contour vous pouvez appeler `stroke(255,255,0)`. Pour l'épaisseur du contour, vous pouvez appeler `strokeWeight(9)`.



BONUS

Exercice 15 Réaliser le dessin à droite. Tous les coups sont permis ! Utilisez toutes les connaissances que vous avez acquises, surtout les figures géométriques via les appels `vertex(...)`, voir l'exercice avec l'étoile du TP précédent.

