

## MU4BM748 – TP Programmation Orientée Objet

### Exercice 1 : cercle, cylindre et cône

- 1) Définir une classe **Cercle()**. Les objets construits à partir de cette classe seront des cercles de tailles variées. En plus de la méthode constructeur (qui utilisera un paramètre **rayon**), définir une méthode **surface()**, qui devra renvoyer la surface du cercle.
- 2) Définir ensuite une classe **Cylindre()**, définie par héritage de la classe **Cercle()**. En plus de la méthode constructeur (qui comportera les paramètres **rayon** et **hauteur**), définir une méthode **volume()**, qui devra renvoyer le volume du cylindre.  
*Rappel* : volume d'un cylindre = surface de section \* hauteur.
- 3) Définir enfin une classe **Cone()**, définie par héritage de la classe **Cylindre()**. En plus de la méthode constructeur (qui comportera également les paramètres **rayon** et **hauteur**), définir une méthode **volume()** propre à cette classe, qui devra renvoyer le volume du cône.  
*Rappel* : volume d'un cône = volume du cylindre correspondant divisé par 3.

### Exercice 2 : jeu de cartes

- 1) Définir une classe **JeuDeCartes()** permettant d'instancier des objets dont le comportement soit similaire à celui d'un vrai jeu de cartes. Cette classe devra comporter au moins les quatre méthodes suivantes :
  - méthode constructeur : création et remplissage d'une liste de 52 éléments, chacun étant un tuple de 2 entiers. Cette liste de tuples contiendra les caractéristiques (valeur et couleur) de chacune des 52 cartes du jeu.
  - méthode **nom\_carte()** : cette méthode doit renvoyer, sous la forme d'une chaîne de caractères, l'identité d'une carte quelconque dont on lui a fourni le tuple descripteur en argument.
  - méthode **battre()** : mélange les éléments de la liste contenant les cartes, quel qu'en soit le nombre.
  - méthode **tirer()** : retire une carte du jeu. Le tuple contenant sa valeur et sa couleur est renvoyé. On retire toujours la première carte de la liste. Si cette méthode est invoquée alors qu'il ne reste plus aucune carte dans le jeu, il faut alors renvoyer l'objet spécial **None**.

Exemple d'utilisation de la classe **JeuDeCartes()** :

```
jeu = JeuDeCartes()
jeu.battre()
for n in range(53):
    c = jeu.tirer()
    if c == None:
        print("Terminé !")
    else:
        print(jeu.nom_carte(c))
```

- 2) Définir deux joueurs A et B. Instancier un jeu de cartes pour chacun des joueurs, et les mélanger. Tirer 52 fois une carte de chacun des deux jeux et comparer leurs valeurs (à chaque fois, la carte la plus forte apporte un point à son joueur). Déterminer le joueur gagnant.