

## Évaluation Dictionnaires et Graphes

Les parties A, B et C sont indépendantes

Franck joue en ligne sur internet. Ce jeu est programmé en Python.

### Partie A

Dans ce jeu Franck incarne un personnage qui doit parcourir des niveaux et tuer des monstres.

Afin de stocker les informations relatives au personnage, le jeu utilise des dictionnaires.

Dont voici un exemple en début de partie :

```
Joueur1= {'nom':'Franck', 'score':0 , 'pv':24}
```

Ici le personnage nommé Franck n'a pas encore ajouté de points à son score et n'a pas encore perdu de point de vie.

1. a. Donner la commande Python permettant d'accéder au point de vie du joueur 1.
- b. Au début du jeu, le joueur trouve une arme permettant d'infliger des dégâts aux monstres qu'il rencontrera plus tard.  
Donner une commande python à entrer dans la console permettant d'ajouter un attribut *degat* au Joueur1 avec une valeur de 3.

```
Joueur1= {'nom':'Franck', 'score':0 , 'pv':24, 'degat':3}
```

Les monstres affrontés par Franck sont également représentés par des dictionnaires.

```
Monstre1 = {'nom':'Fracula', 'degat':2 , 'pv':5}
```

2. À chaque fois qu'un joueur tue un monstre, il marque un point à son score.  
Donner, sans faire de fonction, une commande permettant d'ajouter 1 point à l'attribut score du joueur 1.
3. Pour savoir si un personnage joueur ou monstre est mort, le jeu dispose d'une fonction *est\_mort* qui prend en paramètre un personnage et renvoie un Booléen : vrai si les points de vie du personnage sont inférieur à zéro et faux sinon.  
Donner une implémentation de la fonction *est\_mort*.
4. Lorsqu'un joueur frappe un ennemi, une fonction *frappe* qui prend en paramètre un joueur et un monstre met à jour les points de vie de monstre en enlevant le nombre associé aux dégâts que peut faire le joueur.  
*On ne s'interrogera pas ici sur le cas où les points de vies deviendraient négatifs.*  
Écrire la fonction *frappe*

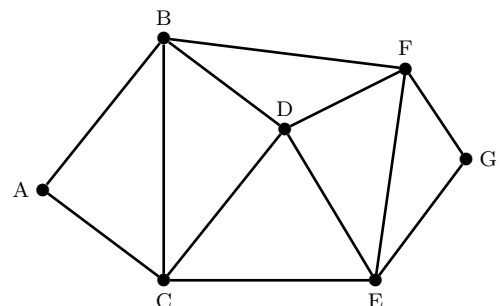
### Partie B

Dans ce jeu vidéo, Franck circule dans des catacombes.

Il a représenté ci-contre le graphe modélisant ces catacombes.

Les sommets représentent les salles et les arêtes représentent les couloirs.

Au début du jeu l'avatar de Franck est dans la salle A.



1. Quel est l'ordre de ce graphe ?
2. Ce graphe est-il connexe ?
3. Lister dans un tableau le degré de chaque sommet.
4. En déduire le nombre d'arêtes qui le compose.
5. Franck souhaite parcourir toutes les salles. Pour cela il souhaite réaliser le parcours en profondeur de ce graphe.  
Donner sans justification les salles qu'il visitera dans l'ordre.  
On prendra les sommets par ordre alphabétique.

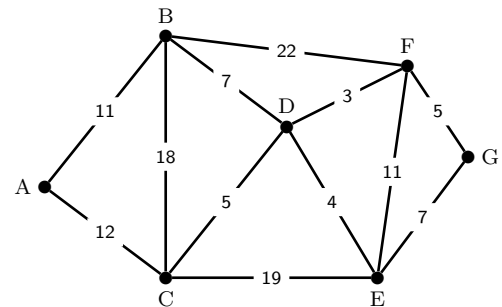
6. Donner la liste d'adjacence de ce graphe.

### Partie C

Les catacombes infestées de monstres qu'il doit combattre.

Après avoir parcouru tous les couloirs, il a rajouté ci-contre l'information suivante :

Les étiquettes du graphe correspondent au nombre de monstres présents dans chaque couloir.



Pour charger les couloirs ainsi que les monstres, il utilise la matrice d'adjacence suivante :

$$A = \begin{pmatrix} 0 & 11 & 12 & 0 & 0 & 0 & 0 \\ 11 & 0 & 18 & 7 & 0 & 22 & 0 \\ 12 & 18 & 0 & 5 & 19 & 0 & 0 \\ 0 & 7 & 5 & 0 & 4 & 3 & 0 \\ 0 & 0 & 19 & 4 & 0 & 11 & 7 \\ 0 & 22 & 0 & 3 & 11 & 0 & 5 \\ 0 & 0 & 0 & 0 & 7 & 5 & 0 \end{pmatrix}$$

On rappelle qu'en python une telle matrice se programme par :

```
A= [[0,11,12,0,0,0,0],[11,0,18,7,0,22,0],[12,18,0,5,19,0,0],[0,7,5,0,4,3,0],[0,0,19,4,0,11,7],[0,22,0,3,11,0,5],[0,0,0,0,7,5,0]]
```

1. Écrire en python la commande permettant de renvoyer le nombre de monstres dans le couloir joignant les salles C et D.
2. On souhaite réaliser une fonction python qui donne le nombre total de monstres présents dans ce niveau du jeu.
  - a. La fonction `nb_monstre` prend en paramètre une matrice et renvoie un entier.  
Donner une implémentation de cette fonction.
  - b. Que retourne l'instruction `nb_monstre(A)` ?
3. Franck débute le jeu dans la salle A et doit atteindre l'adversaire final en salle G.  
Il souhaite, pour ce faire, croiser le moins de monstres possible.  
En exécutant un algorithme, déterminer ce trajet minimal et préciser le nombre de monstres affrontés.