

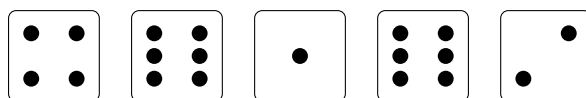
Devoir surveillé n°3

Jeu de Yam, durée 1h

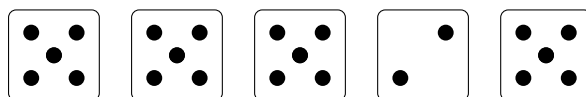
Au jeu de Yam on jette cinq dés à six faces, dans le but d'obtenir certaines figures.

La figure la plus forte est le Yam : tous les dés sont identiques. Ensuite vient le carré : quatre dés sont identiques. Il existe aussi la suite, le full, le brelan, la paire, la double paire, etc.

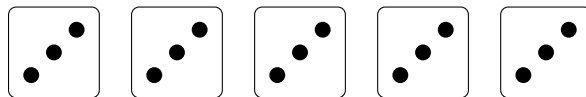
Voici par exemple une paire de six :



Un carré de cinq :



Un Yam de trois :



Question 1

Ecrire la commande dans l'éditeur permettant d'importer la fonction `randint` du module `random`.

La commande `randint(a,b)` renvoie un nombre pseudo-aléatoire compris entre les entiers a et b inclus.

Question 2

Dans l'interpréteur (*i.e.*, le shell), préciser la commande permettant de simuler un dé, c'est-à-dire renvoyant un entier compris entre 1 et 6.

Question 3

Ecrire une fonction `Tirage(D)` simulant un tirage de D dés, c'est-à-dire renvoyant une liste de D entiers compris entre 1 et 6.

A. Compteur de Yams et de carrés**Question 4**

Écrire une fonction `Occ(L)` qui reçoit une liste L et compte le nombre de 1, de 2, etc jusqu'à 6 que contient la liste L . Cette fonction renverra donc une liste de 6 entiers. Si par exemple $T=[4,2,4,4,5]$ alors on doit obtenir $Occ(T)=[0,1,0,3,1,0]$.

Question 5

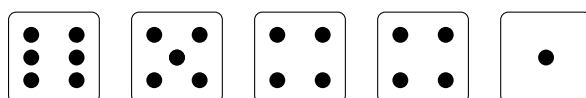
Écrire une fonction `Max(L)` qui reçoit une liste non-vide L de nombres et renvoie son plus grand élément. L'utilisation de la fonction `max()` intrinsèque de Python n'est pas permise.

Question 6

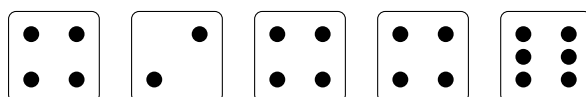
En utilisant les fonctions précédentes, écrire un programme qui simule N tirages ($N=1000$) de cinq dés (sans l'afficher) et affiche «Yam» ou «Carré» si on obtient un Yam ou un carré. Dans le cas où un Yam ou un carré est obtenu on affichera le tirage obtenu. Ajouter un compteur du nombre de Yams et du nombre de carrés, et afficher à la fin une phrase correcte informant l'utilisateur du nombre de Yams et de carrés obtenus.

B. Jeu à trois tirages

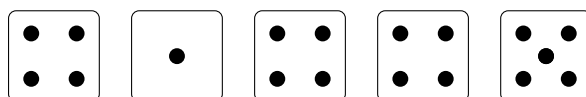
Selon la règle du Yam, on peut relancer deux fois certains des dés obtenus. Par exemple si on obtient



alors on relance les dés ayant donné 6, 5, et 1. On obtient alors



On relance les dés ayant donné 2 et 6. On obtient



On a finalement obtenu un brelan (trois dés identiques), on ne peut plus continuer.

On souhaite maintenant simuler ce procédé. Le but est d'obtenir un Yam, donc on choisira toujours de garder les dés qui apparaissent en plus grand nombre.

N.B. Il est possible de relancer lors du troisième tirage un dé qui a été écarté lors du second tirage.

Question 7

Reprendre la fonction **Max** pour créer une fonction **MaxInd** qui reçoit une liste de nombres L et renvoie l'indice du plus grand élément de cette liste (c'est-à-dire sa position dans L). Si le maximum apparaît en plusieurs endroits alors la fonction renverra l'un de ses indices. La fonction devra être de complexité linéaire.

Par exemple on pourrait tester dans l'interpréteur :

```
>>> MaxInd([0,3,10,4,0,2,10,9])
2
```

En effet, le plus grand élément est 10, et il apparaît en position 2. La fonction peut aussi renvoyer 6.

Question 8

À l'aide des fonctions **Occ** et **MaxInd** créer un programme qui simule un tirage de cinq dés, et qui relance les dés qui ne sont pas égaux au numéro le plus souvent apparu. On obtient ainsi un second tirage, qui sera aussi affiché.

L'exécution de ce programme peut donner par exemple

```
[6,5,4,4,1]
[4,2,4,4,6]
```

Question 9

Que faudrait-il faire pour ajouter un troisième tirage ?

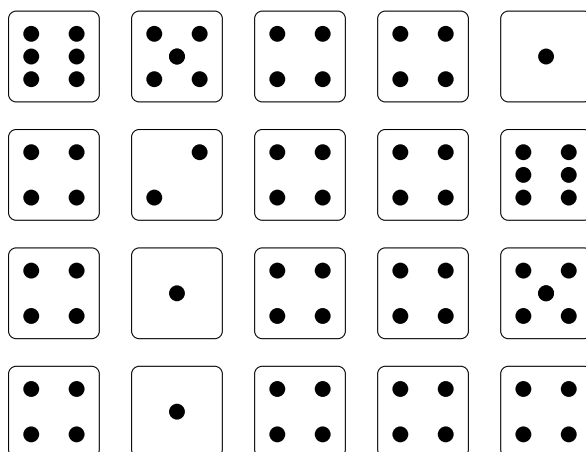
C. Nombre de tirages nécessaires pour obtenir un Yam

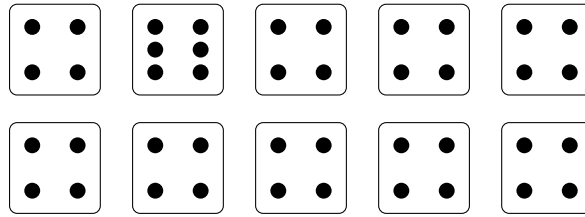
On suppose que l'on a une copie du programme précédent.

Question 10

Modifier ce programme afin qu'il continue à jeter les dés jusqu'à ce qu'il obtienne un Yam. Compter le nombre de tirages nécessaires.

Par exemple si les tirages sont :





Alors il aura fallu 6 tirages pour obtenir un Yam.