

Fonction randint

Fonction choice

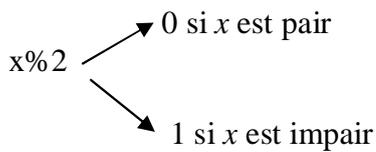
Notions à connaître en Python pour les simulations de variables aléatoires :

$a\%b$: reste de la division euclidienne de a par b (a entier relatif quelconque, b entier naturel non nul)

Exemple :

$14\%3$ donne 2

x entier relatif quelconque



Le 29-1-2024

On lance deux dés cubiques non truqués dont les faces sont numérotées de 1 à 6.

Le joueur lance les deux dés.

Si les deux numéros sont pairs il gagne 3 €

Si les deux numéros sont impairs, il gagne 2 €

Dans les autres cas, il ne gagne ni ne perd rien.

Écrire un programme Python qui simule une partie de ce jeu.

Solution :

On importe la fonction randint de la bibliothèque random.

```
def jeu():
    x,y=randint(1,6), randint(1,6)
    if x%2==0 and y%2==0:
        return 3
    elif x%2==1 and y%2==1:
        return 2
    else:
        return 0
```

Le dimanche 21-4-2024

Source : T spé interrogation écrite du 9-5-2023

Une urne contient 3 boules bleues, 2 boules rouges et 5 boules jaunes. Les boules sont toutes distinguables, numérotées par exemple. On tire successivement et avec remise 4 boules dans l'urne.

Le but de cette question est de simuler n tirages successifs aléatoires avec remise dans l'urne, n étant un entier naturel $n \geq 1$.

On considère la fonction Python d'en-tête `def tirage(n)` : écrite dans le cadre ci-dessous qui prend pour argument un entier naturel $n \geq 1$ et dont le but est de renvoyer une liste correspondant à n tirages successifs avec remise dans l'urne.

La liste `U` qui apparaît sur la deuxième ligne représente le contenu de l'urne. Chaque boule est désignée par une lettre correspondant à sa couleur : `B` pour bleue, `R` pour rouge et `J` pour jaune.

```
def tirage(n):
    U=['B']*3+['R']*2+['J']*5
    L=[]
    for i in range(n):
        r=randint(0, 9)
        L.append(U[r])
    return L
```

On suppose que la fonction `randint` a été préalablement importée de la bibliothèque `random`. On rappelle que `randint(a, b)` choisit un entier aléatoire entre a et b (compris).

On peut faire puis tester le programme Python sur calculatrice.

On notera la syntaxe `U=['B']*3+['R']*2+['J']*5` qui remplace `U=['B', 'B', 'B', 'R', 'R', 'J', 'J', 'J', 'J', 'J']`.

On peut aussi écrire `U=3*['B']+2*['R']+5*['J']`.

On peut proposer d'autres versions en utilisant la fonction `choice` préalablement importée de la bibliothèque `random`.

```
def tirage(n):
    U=['B']*3+['R']*2+['J']*5
    L=[]
    for i in range(n):
        b=choice(U)
        L.append(b)
    return L
```

Autre possibilité très simple :

```
def tirage(n):  
    U=['B']*3+['R']*2+['J']*5  
    L=[choice(U) for i in range(n)]  
    return L
```