

**TP 4 – suite**  
**Boucle WHILE**

Dans ce TP, on pourra utiliser les fonctions de la bibliothèque `random` :

- `randrange(n)` : renvoie un nombre aléatoire (loi uniforme) entre 1 et  $n - 1$ . ( $n \in \mathbb{N}^*$ )
- `randrange(a,b)` : renvoie un nombre aléatoire (loi uniforme) entre  $a$  et  $b - 1$ ,  $a$  et  $b$  sont deux entiers tels que  $a < b$
- `randint(a,b)` : renvoie un nombre aléatoire (loi uniforme) entre  $a$  et  $b$

**Exercice 1** On effectue des lancers successifs d'un dé cubique non truqué.

1. Comment simuler en langage `python` un lancer de dé?
2. Écrire une fonction `jeu(valeur)` qui prend en argument une variable `valeur` (un entier compris entre 1 et 6) et qui renvoie le nombre de lancers nécessaires pour obtenir `valeur`.

**Exercice 2** Un rat de laboratoire est soumis à l'expérience suivante:

il est enfermé dans une cage comportant quatre portes, derrière lesquelles se trouve un morceau de gruyère. Trois des quatre portes sont munies d'un dispositif envoyant au rat une décharge électrique s'il essaye de les franchir; la quatrième laisse le passage libre.

1. Comment simuler un essai du rat ?
2. Dans cette question, on suppose que le rat abandonne au bout de 20 essais.  
Écrire une fonction Python qui simule l'expérience aléatoire et renvoie
  - le nombre d'essais effectués par le rat s'il trouve la bonne porte au bout des 20 essais.
  - 0 s'il ne trouve pas la bonne porte au bout des 20 essais.
3. Dans cette question, on suppose que le rat n'abandonne jamais. Écrire une fonction Python qui simule l'expérience aléatoire et renvoie le nombre d'essais effectués par le rat jusqu'à ce qu'il trouve la bonne porte.