



DM 3

Informatique

Des dés pour DD

Simon Dauguet
simon.dauguet@gmail.com

Pour le Vendredi 06 Mars 2026

On renverra le code dans un fichier dont le nom devra être "DM3.Nom.Prénom.py" (attention, l'extension n'est pas toujours affiché par défaut sur les machines fonctionnant sous windows (c'est nul windows)).

On dispose d'un jeu de dés équilibrés, qu'on lance l'un après l'autre. On appelle *résultat* la somme des faces obtenues.

On modélise un jeu de dés par la liste du nombre de faces de dés. Par exemple, le jeu de dés `[6, 6, 4]` est constitué de deux dés à 6 faces et d'un dé à 4 faces. Il permet d'obtenir des résultats de 3 à 16.

Étant donné un jeu de dés, on modélise un lancer par la liste des faces obtenues. Par exemple, un lancer possible avec le jeu `[6, 6, 4]` est `[3, 5, 1]`, ce qui signifie que l'on a obtenu un 3, un 5 et un 1 lors du lancer. Le résultat obtenu est alors $3 + 5 + 1 = 9$.

On souhaite connaître les probabilités des différents résultats possibles.

1 Généralités

1. Écrire une fonction `rajouter_lancer(lancers:list, de:int) -> list` qui complète une liste de lancers par tous les lancers possibles d'un à `de` faces. Par exemple, `rajouter_lancers([[3, 5], [6, 1]], 4)` (on a déjà lancé deux dés, et on lance un troisième dé à 4 faces, renvoie, à l'ordre près :
`[[3, 5, 1], [3, 5, 2], [3, 5, 3], [3, 5, 4], [6, 1, 1], [6, 1, 2], [6, 1, 3], [6, 1, 4]]`.
2. Écrire une fonction `liste_lancers(jeu:list) -> list` qui renvoie la liste de tous les lancers possibles avec le jeu de dés `jeu`. Par exemple, `liste_lancers([4, 2])` (on lance un dé-4 et un dé-2) renvoie, à l'ordre près :
`[[1, 1], [1, 2], [2, 1], [2, 2], [3, 1], [3, 2], [4, 1], [4, 2]]`.
3. Écrire une fonction `resultats(jeu:list) -> list` qui renvoie la liste des résultats (c'est-à-dire des sommes) correspondante, sans doublons. Par exemple, `resultats([4, 2])` renvoie `[2, 3, 4, 5, 6]`.
4. Écrire une fonction `occurrences(jeu:list) -> list` qui renvoie la liste des couples des résultats possibles et de leur nombre d'apparition. Par exemple `occurrences([4, 2])` renvoie `[(2, 1), (3, 2), (4, 2), (5, 2), (6, 1)]` (car il n'y a qu'une seule façon d'avoir le résultat 2, deux façons d'avoir le résultat 3, deux façons d'avoir le résultat 4, deux façons d'avoir le résultat 5, et une façon d'avoir le résultat 6).
5. Modifier la fonction `occurrences` pour faire une fonction `proba(jeu:list) -> list` qui renvoie la liste des couples des résultats possibles et leur probabilité respectives.
6. Écrire une procédure `representation(jeu:list) -> None` qui affiche un graphe en barre avec les résultats possibles en abscisses et leur probabilités correspondantes en ordonnée.
7. Écrire une fonction `proba_cumul_inv(jeu:list) -> list` qui renvoie la liste des couples des résultats et de leur probabilités cumulées inverses, c'est-à-dire `jeu[k]` est un couple, `jeu[k][0]` un résultat possible et `jeu[k][1]` est la probabilité d'avoir un résultat au moins égal à `jeu[k][0]`.
8. Écrire une procédure `rep_cumul(jeu:list) -> None` qui affiche les probas cumulés inverses associées au jeu de dés `jeu`.

2 Applications à un jeu de plateau

Dans cette partie, on se pose la question inverse : que jeu de dés est quel résultat nous faut-il pour simuler une probabilité donnée ?

On suppose que l'on dispose d'un nombre illimité de dés 4, 6, 8, 12 et 20 faces. On note : `types_des = [4, 6, 8, 12, 20]` une variable globale.

9. En vous inspirant des questions 1 et 2 :

(a) Écrire une fonction `rajouter_de(jeux:list)` -> `list` qui prend en argument une liste de jeux de dés, et rajoute tous les dés possibles à ces jeux de dés. Par exemple, `rajouter_de([[4,6],[8,4]])` (donc deux jeux de dés `[4,6]` et `[8,4]`) renvoie, à l'ordre près :

`[[4,6,4],[4,6,6],[4,6,8],[4,6,12],[4,6,20],[8,4,4],[8,4,6],[8,4,8],[8,4,12],[8,4,20]]`.

(b) Écrire une fonction `liste_jeux(n:int)` -> `list` qui renvoie la liste des jeux de `n` dés possibles. Par exemple, `liste_jeux(3)` renvoie une liste de longueur $5^3 = 125$.

10. Écrire une fonction `sol(p:int, q:int)` -> `tuple` qui prend en arguments deux entiers $1 \leq p \leq q$ et qui renvoie les couples `(jeu,k)`, où `jeu` est une liste de jeu de dés de 8 dés ou moins, répondant au problème : "la probabilité d'obtenir `k` ou moins en lançant le jeu de dés `jeu` est égale à `p/q`".

11. Répondre au meme :

