

Fluctuations d'une fréquence selon les échantillons, probabilités

Deux personnes jouent au « **jeu du 7** » qui consiste, en lançant un ou deux dés, à obtenir « 7 points » pour gagner la partie.

Pour chaque lancer, une des deux personnes utilise un dé à 12 faces sur lequel il lit les points indiqués sur la face supérieure et l'autre utilise deux dés à 6 faces pour lesquels il additionne les points indiqués sur les faces supérieures des deux dés.



Problématique : les chances des deux joueurs sont-elles les mêmes ?

- Reformuler cette problématique en termes de probabilités :

.....
.....

- Proposer une méthode qui permettrait de répondre à la problématique posée :

.....
.....
.....
.....

- Choisir parmi les quatre propositions suivantes deux fonctions permettant de simuler les lancers d'un dé à 12 faces et de deux dés à 6 faces :

```
def lancer1():  
    return randint(1,6) + randint(1,6)  
def lancer2():  
    return randint(1,12)  
def lancer3():  
    return 2*randint(1,6)  
def lancer4():  
    return randint(1,12) + randint(1,12)
```

Valider votre choix en réalisant plusieurs essais.

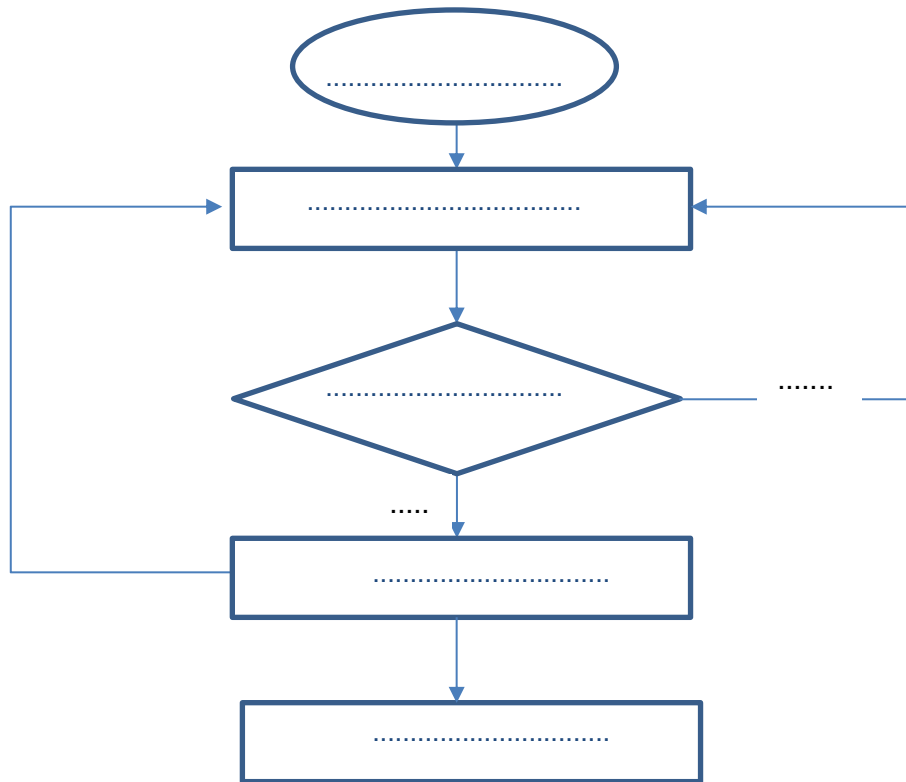
```
from random import*  
def lancer1():  
    .....  
print(lancer1())
```

Fluctuations d'une fréquence selon les échantillons, probabilités

- Après avoir choisi les deux fonctions permettant de simuler les lancers d'un dé à 12 faces et de deux dés à 6 faces, écrire un algorithme puis un script permettant de simuler 10 lancers consécutifs

Algorithme	Pseudo-code
.....
.....
.....
.....
.....
.....
.....
.....
.....
.....

Algorithme :



Scripts :

<pre> # Lancer avec 1 dé 12 faces def un_de(): for i in range(.....): print(r) if r==7: print("c= ",c) return c/n </pre>	<pre> # Lancer avec 2 dés 6 faces def deux_des(): for i in range(.....): print(r) if r==7: print("c= ",c) return c/n </pre>
--	---

- Créer une fonction unique traitant simultanément les deux cas :

.....

Fluctuations d'une fréquence selon les échantillons, probabilités

- Créer une fonction unique traitant simultanément les deux cas :

```
def deux_cas():
    c=0
    d=0
    for i in range(10):
        r=lancer2()
        if r==7:
            c=c+1
    print("fréquence avec 1 dé ",c/n)
    for i in range(10):
        r=lancer1()
        if r==7:
            d=d+1
    print("fréquence avec 2 dés ",d/n)
```

```
print("Deux cas ", deux_cas())
```

- Réponse à la problématique :

.....

.....

.....

.....

Modifier dans le script la valeur du nombre de lancers simulés :

```
def deux_cas(n):
    c=0
    d=0
    for i in range(n):
        r=lancer2()
        if r==7:
            c=c+1
    print("fréquence avec 1 dé ",c/n)
    for i in range(n):
        r=lancer1()
        if r==7:
            d=d+1
    print("fréquence avec 2 dés ",d/n)
```

```
n=int(input("Nombre de lancers ? "))
print("Deux cas ", deux_cas(n))
```

Conclusion :

Avec 1 dé à 12 faces la fréquence de 7 obtenu tend vers

Avec 2 dés à 6 faces la fréquence de 7 obtenu tend vers

.....

.....

.....

.....

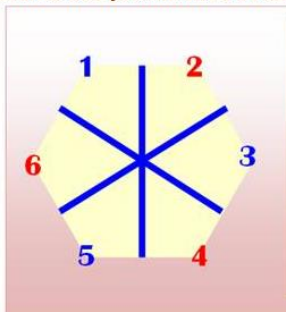
.....

.....

Fluctuations d'une fréquence selon les échantillons, probabilités

➤ Calcul des probabilités :

Résumé des probabilités avec le lancer d'un seul dé



P_n = Probabilités d'avoir le nombre n

$$P_1 = 1/6; P_2 = 1/6; P_3 = 1/6$$

$$P_4 = 1/6; P_5 = 1/6; P_6 = 1/6$$

$$P_{\neq 1} = 5/6; P_{\neq 2} = 5/6; P_{\neq 3} = 5/6$$

$$P_{\neq 4} = 5/6; P_{\neq 5} = 5/6; P_{\neq 6} = 5/6$$

$$P_{1 \text{ ou } 2} = 1/3; P_{1 \text{ ou } 3} = 1/3; P_{1 \text{ ou } 4} = 1/3 \dots$$

$$P_{1 \text{ ou } 2 \text{ ou } 3} = 1/2; P_{1 \text{ ou } 2 \text{ ou } 4} = 1/2 \dots$$

$$P_{\text{Pair}} = 1/2; P_{\text{Impair}} = 1/2$$

$$P_{>1} = 5/6; P_{>2} = 2/3; P_{>3} = 1/2 \dots$$

Toutes les possibilités avec deux dés

- Le tableau à double-entrée montre les 36 = 6 x 6 possibilités

11	21	31	41	51	61
12	22	32	42	52	62
13	23	33	43	53	63
14	24	34	44	54	64
15	25	35	45	55	65
16	26	36	46	56	66

<http://villemain.gerard.free.fr/Wwwgvm/Probabil/DesDeux.htm#D2>

Quelle est la probabilité d'obtenir un 7 avec un dé 12 faces et deux dés 6 faces ?

.....

.....

.....

.....