Il faut passer les premiers !

D'après un sujet du concours général.

Pour faire une réussite, on dispose d'un jeton, d'un dé équilibré à six faces et de la grille ci-contre :
Le joueur commence la réussite en posant le jeton sur la case 0.
Il fait ensuite une série de lancers du dé, lorsque le dé affiche la valeur k, il avance le jeton de k cases et :
– s'il atteint ou dépasse la case numéro 24, il a gagné ;
– s'il arrive à une case dont le numéro est un nombre premier inférieur à 24, il a perdu et le jeu s'arrête ;
– dans les autres cas, il relance le dé et continue la réussite.

0	1	2	3	4
9	8	7	6	5
10	11	12	13	14
19	18	17	16	15
20	21	22	23	24

On s'intéresse à la probabilité de gagner à cette réussite.

1. a) Quelle est la probabilité de perdre dès le premier lancer ?
   b) Quelle est la probabilité de perdre au deuxième lancer ?

2. a) Quelle est la probabilité que la réussite s'achève sur la case 2 ?
   b) Quelle est la probabilité qu'elle s'achève sur l'une des cases de la première ligne ?

3. Un joueur a écrit l'algorithme ci-contre.
   a) Quel est le rôle de la boucle « tant que » ?
   b) Quel est le rôle de l'instruction :
      Pion=Pion + ENT(6*Alea()+1) ?
   c) Quel est le rôle de la variable S ?
   d) Quelle est l'utilité de cet algorithme pour le joueur ?

4. Déterminer la valeur exacte ou, à défaut, une estimation de la probabilité de gagner à cette réussite.

Il faut passer les premiers !
Eléments de correction

On s'intéresse à la probabilité de gagner à cette réussite.

1. a) Quelle est la probabilité de perdre dès le premier lancer ?
   b) Quelle est la probabilité de perdre au deuxième lancer ?

a) Les tirages possibles, équiprobables, sont 1, 2, 3, 4, 5 et 6.
Les résultats 2, 3 et 5 sont perdants, la probabilité de perdre au premier lancer est donc 3/6 = 1/2.

b) On perd au deuxième lancer si on obtient : (1,1), (1,2), (4,3) ou (6,5) soit 4 cas sur 36, la probabilité de perdre au deuxième lancer est donc de 4/36 = 1/9.

2. a) Quelle est la probabilité que la réussite s'achève sur la case 2 ?
   b) Quelle est la probabilité qu'elle s'achève sur l'une des cases de la première ligne ?

a) Deux possibilités : obtenir 2 au premier lancer ou 1 puis 1 au deuxième lancer.
La probabilité de cet événement est de 1/2+ 1/36 = 7/36
b) La réussite s'achève sur la première ligne si l'on arrive sur la case 2 ou sur la case 3, les cas possibles sont donc (2), (3), (1,1) ou (1,2), la probabilité est donc de 2/6 + 2/36 = 7/18.

3. Un joueur a écrit l'algorithme ci-contre.
a) Quel est le rôle de la boucle « tant que » ?
b) Quel est le rôle de l'instruction :
Pion <-- Pion + ENT(6*Alea()+1) ?
c) Quel est le rôle de la variable S ?
d) Quelle est l'utilité de cet algorithme pour le joueur ?

a) Chacun des passages dans cette boucle simule une partie : tant que pion n'a pas atteint la case 24 et n'est pas tombé sur un entier, le jeu continue. La boucle est répétée N fois.
b) La formule ENT(6*ALEA()+1) permet d'obtenir un nombre aléatoire entier entre 1 et 6, elle simule donc le lancer d'un dé, on déplace donc le pion du nombre de cases correspondant.
c) La variable S compte ne nombre de parties gagnées sur N parties simulées.
d) L'algorithme donne en sortie la fréquence des parties gagner, cela permet au joueur d'obtenir une estimation de la probabilité de gagner.

4. Déterminer la valeur exacte ou, à défaut, une estimation de la probabilité de gagner à cette réussite.

Remarque : La résolution de la cette question est longue et demande de l'organisation, elle est donc conçue pour être un travail collaboratif pour la partie académique du concours qui est un travail de groupe, une partie du groupe pouvant se charger de la programmation de l'algorithme de la question 3.

Un travail « à la main » peut se faire à partir d'arbres répartis en plusieurs morceaux … il est peu probable d'arriver au bout de se travail, on prend donc en compte l'organisation et les résultats partiels.

Estimation à l'aide d'un programme sur calculatrice :

```
from math import*
from random import*
def freqSucces(N) :
    LPremier=[2,3,5,7,11,13,17,19,23]          # liste des nombres premiers inférieurs à 24
    S=0
    for k in range(N) :
        Pion=0
        Perdu=0
        while Pion<24 and Perdu==0 :
            Pion=Pion + floor(6*random()+1)     # random() renvoie un nombre aléatoire de [0;1[
                if Pion in LPremier :
                    Perdu=1
                if Perdu==0 :
                    S=S+1
    return(S/N)
```

Des exemples d'exécution de ce programme :

```
>>> freqSucces(10000)
0.0607

>>> freqSucces(10000)
0.0601

>>> freqSucces(10000)
0.0604
```

on peut estimer la probabilité de gagner à environ 6 %.