69 Numérotation en base 16

Dans le système de numération en base seize, les caractères utilisés sont les chiffres de 0 à 9 et les lettres de A à F.

Un nombre de ce système de numération est une suite de plusieurs caractères de ce type. Par exemple, on peut considérer les nombres A5F, 1A, 331, AB, C81C, ...

Remarques :

- un nombre de plusieurs caractères ne peut pas commencer par zéro ;
- le nombre qui s'écrit 1A en base seize a pour valeur décimale $1 \times 16 + 10$, soit 26.

On considère les nombres de deux caractères écrits en base seize.

1. Montrer qu'il y a 240 nombres de deux caractères en base seize. On pourra s'aider de l'ébauche d'un arbre.

2. On écrit au hasard un nombre de deux caractères en base seize. On considère les événements suivants :

- E_1 « le nombre ne contient aucune lettre » ;
- E_2 « le nombre commence par 1 ».

a. Déterminer les probabilités de E_1 et de E_2.

b. Déterminer la probabilité de l'événement $E_1 \cap E_2$.

c. Déterminer la probabilité de l'événement $E_1 \cup E_2$.

d. Déterminer la probabilité pour que le nombre contienne au moins une lettre.

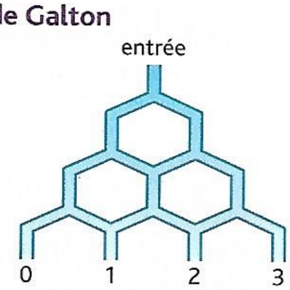
e. Déterminer la probabilité pour que le nombre soit formé de deux caractères différents.

Le saviez-vous ?

Le système de numération en base 16, appelé également système hexadécimal, est très utilisé en informatique : il permet notamment d'écrire de façon plus compacte les codes binaires utilisés par les ordinateurs. C'est un des modes de codage informatique des couleurs des écrans d'ordinateurs.

70 algo Planche de Galton

On considère l'expérience suivante : on lâche une bille en haut de la planche ci-contre (entrée). À chaque intersection, elle a autant de chances de tomber à droite qu'à gauche. On s'intéresse au numéro de sortie de la bille en bas de la planche.



1. a. On souhaite simuler un lâcher de bille. On propose l'algorithme suivant :

```
Variables :
    a, b, c : entiers ;
Début
    a ←EntierAléaEntre(0 ; 1) ;
    b ←EntierAléaEntre(0 ; 1) ;
    c ←EntierAléaEntre(0 ; 1) ;
    Afficher(« le numéro de sortie est : », a + b + c) ;
Fin.
```

Expliquer la démarche.

b. Mettre en place la feuille de calcul suivante :

A2 | fx =ENT(2*ALEA())+ENT(2*ALEA())+ENT(2*ALEA())

	A	B	C	D	E	F	G
1	numéro de sortie de la bille		numéro	0	1	2	3
2	2		Effectif sur 100 simulations				
3	3		Effectif sur 1000 simulations				
4	2						
5	1						

Compléter les cellules D2, E2, etc. à l'aide de la fonction NB.SI.

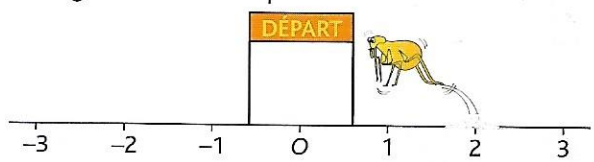
c. Estimer la probabilité de chaque numéro de sortie possible. Expliquer la démarche.

2. Construire un arbre décrivant tous les chemins possibles de la bille, puis calculer la probabilité de chaque numéro de sortie de la bille.

Comparer avec les résultats obtenus à la question **1.**

71 algo Marche aléatoire

Une puce se déplace sur un axe gradué. À chaque pas, de façon aléatoire, elle avance ou recule d'une unité. Elle part de l'origine O et effectue quatre sauts.



1. Quelles sont les abscisses finales possibles de la puce ?

2. a. On propose l'algorithme suivant pour simuler le premier saut de la puce :

```
Variables :
    x, N : entier ;
Début :
    x ← 0 ;
    N ←EntierAléaEntre(0 ; 1) ;
    Si N = 0 alors x ← x − 1 ; sinon x ← x + 1 ;
    FinSi ;
    Afficher(« l'abscisse après un saut est » , x) ;
Fin.
```

Expliquer.

b. Modifier l'algorithme de façon à simuler les quatre sauts de la puce.

3. a. Programmer à l'aide d'une calculatrice, d'un tableur ou d'un logiciel.

b. À l'aide du programme, dire si les affirmations suivantes semblent justes ou fausses (on justifiera) :

i. il existe une position finale de la puce plus probable que les autres ;

ii. la puce se trouve à la fin en O avec une probabilité de 0,5 ;

iii. la puce a autant de chance d'avoir à la fin une abscisse positive qu'une abscisse négative.

4. Construire un arbre décrivant tous les sauts possibles de la puce, puis calculer la probabilité de chaque abscisse finale possible.

Comparer avec les résultats obtenus à la question **3.**