

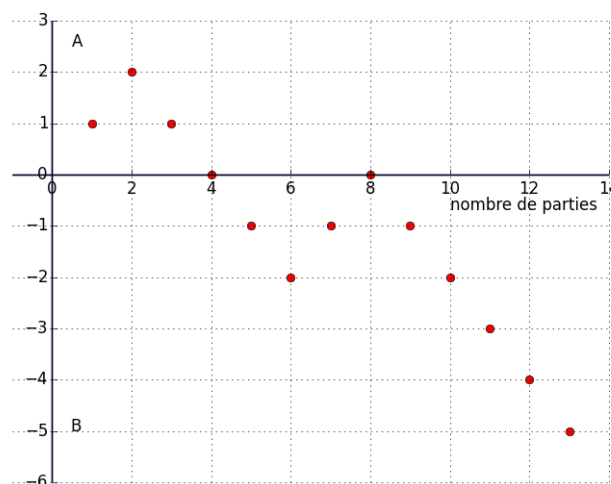
Deux joueurs A et B s'affrontent dans un jeu vidéo. Un joueur marque +1 s'il gagne une partie, -1 s'il perd. Le vainqueur du duel est le premier qui arrive à 5.

On suppose qu'à chaque partie, le joueur A possède trois chances sur 5 de gagner. Le déroulement du jeu peut être assimilé au déplacement aléatoire d'un pion sur un axe vertical : au départ, le pion est à l'origine : si A gagne, le pion monte d'un cran ; si B gagne il descend d'un cran.

1 Simulation du jeu

1.1 Exemple de marche aléatoire

Le graphique ci-contre représente une marche aléatoire associée à un duel.



Q1) Qui a gagné ce duel ? Après combien de parties.

Q2) Quel est le nombre de parties gagnées par A ?

Q3) Combien de fois est-il repassé par l'origine ? Interprétez ces éventualités.

1.2 Nombre de parties jouées

On note N la variable aléatoire qui indique le nombre de parties jouées jusqu'à ce qu'il y ait un gagnant.

Q4) Quelle est la plus petite valeur que N peut prendre ? Avec quelle probabilité ?

Q5) Justifier que N peut prendre une infinité de valeurs.

1.3 Algorithme de simulation du jeu

Voici un algorithme, à compléter dans le fichier `marche_alea_partieAeleve.alg`, permettant de représenter un duel et d'indiquer le nombre de parties nécessaires pour obtenir un vainqueur.

- y désigne l'ordonnée du pion ;
- N désigne le nombre de parties jouées

Q6) Quel est le rôle de la variable h ?

Q7) A quelle condition le duel s'arrête-t-il ? Déduisez-en le critère à inscrire à la ligne 8.

Q8) Testez cet algorithme plusieurs fois. Existe-t-il toujours un vainqueur du duel ?

```

1: VARIABLES
2: h EST_DU_TYPE NOMBRE
3: y EST_DU_TYPE NOMBRE
4: N EST_DU_TYPE NOMBRE
5: DEBUT_ALGORITHME
6:   N PREND_LA_VALEUR 0
7:   y PREND_LA_VALEUR 0
8:   TANT_QUE (.....) FAIRE
9:     DEBUT_TANT_QUE
10:      h PREND_LA_VALEUR ALGOBOX_ALEA_ENT(1,5)
11:      SI (.....) ALORS
12:        DEBUT_SI
13:          y PREND_LA_VALEUR y+1
14:        FIN_SI
15:      SINON
16:        DEBUT_SINON
17:          y PREND_LA_VALEUR .....
18:        FIN_SINON
19:      N PREND_LA_VALEUR N+1
20:      TRACER_POINT (N,y)
21:    FIN_TANT_QUE
22:    AFFICHER "Le nombre de parties est : "
23:    AFFICHER N
24: FIN_ALGORITHME

```

Les instructions de la ligne 6 à la ligne 21 privées de la ligne 20 (utilisée pour le graphique), constituent le programme « nombre de parties de la partie C ».

2 Fréquence de réalisation de l'événement « A est vainqueur »

On note S_A l'événement « A est vainqueur » lors d'un duel.

On se propose de simuler 10 000 duels et d'afficher la fréquence de réalisation de l'événement S_A .

On admet qu'on peut appliquer l'intervalle de confiance $\left[f - \frac{1}{\sqrt{n}}; f + \frac{1}{\sqrt{n}} \right]$ vu en seconde et que la fréquence f observée a 95 % de chances d'être une approximation à $\frac{1}{\sqrt{10000}} = 10^{-2}$ près de la probabilité p de S_A .

On donne ci-dessous (voir le fichier `marche_alea_partieBeleve.alg`) un algorithme qui affiche la fréquence de réalisation de l'événement S_A sur un échantillon de 10000 lancers.

- La variable L est une liste dans laquelle on enregistre les 10000 résultats d'un duel.
- La variable k est un compteur qu'on incrémente à chacun des 10000 duels.
- S est la variable qui indique le nombre de réalisations de l'événement S_A .
- F est la variable qui donne la fréquence de S_A .

```

1: VARIABLES
2: h EST_DU_TYPE NOMBRE
3: y EST_DU_TYPE NOMBRE
4: N EST_DU_TYPE NOMBRE
5: k EST_DU_TYPE NOMBRE
6: F EST_DU_TYPE NOMBRE
7: S EST_DU_TYPE NOMBRE
8: DEBUT_ALGORITHME
9:   S PREND_LA_VALEUR 0
10:  POUR k ALLANT_DE 1 A 10000
11:    DEBUT_POUR
12:    y PREND_LA_VALEUR 0
13:    TANT_QUE (.....) FAIRE
14:      DEBUT_TANT_QUE
15:      h PREND_LA_VALEUR ALGOBOX_ALEA_ENT(1,5)
16:      SI (.....) ALORS
17:        DEBUT_SI
18:        y PREND_LA_VALEUR y+1
19:        FIN_SI
20:      SINON
21:        DEBUT_SINON
22:        y PREND_LA_VALEUR .....
23:        FIN_SINON
24:      FIN_TANT_QUE
25:    SI (y==5) ALORS
26:      DEBUT_SI
27:      S PREND_LA_VALEUR .....
28:      FIN_SI
29:    FIN_POUR
30:  F PREND_LA_VALEUR .....
31:  AFFICHER "La fréquence de S_A est : "
32:  AFFICHER F
33: FIN_ALGORITHME

```

Q9) Le bloc (ligne 11 à 23) reprend le programme précédent qui simule un duel. Complétez les instructions qui manquent.

Q10) Quel est le rôle de l'instruction conditionnelle (ligne 24 à 27) ? Complétez-la.

Q11) Quelle valeur prend la variable F (ligne 29) ?

Q12) Testez ce programme. Estimez la probabilité de S_A .

Q13) Comment modifier l'algorithme pour obtenir une estimation de la probabilité de l'événement S_B , « B est vainqueur lors d'un duel » ?

Q14) Peut-on affirmer *a priori* que les événements S_A et S_B sont des événements contraires ?

3 Nombre moyen de parties dans un duel

On se propose de calculer le nombre moyen de parties lors de la simulation de 10000 duels. On donne ci-contre le programme de cet algorithme.

Variables :

L(liste) h,k,y,N,T,F (nombres)

Entrée:

T reçoit 0

Traitement:

Pour k variant de 1 à 10000

Programme "nombre de parties"

Enregistrer N dans L[k]

T reçoit T+L[k]

FinPour

F reçoit T/10000

Sortie:

Afficher "moyenne du nombre de parties"

Afficher F

- L est la liste où on enregistre le nombre N de parties lors d'un duel

- h, k, y sont les variables utilisées en partie B

- T est la variable qui indique la somme des éléments de la liste L

- F est la variable qui indique la moyenne des parties sur les 10000 duels.

Q15) Ecrivez ce programme avec Algobox. Testez-le.

Q16) Donnez une estimation du nombre moyen de parties durant ce duel.

4 Réponses

Réponse Q1): $\frac{1}{2}$ Réponse Q2): $\frac{1}{2}$ Réponse Q3): $\frac{1}{2}$ Réponse Q4): $880'000 \approx \left(\frac{5}{3}\right) + \left(\frac{5}{2}\right)$ Réponse Q8): $\frac{1}{2}$

Réponse Q9): $\frac{1}{2}$ Réponse Q12): $\frac{1}{2}$

Réponse Q13): $\frac{1}{2}$

Réponse Q14): $\frac{1}{2}$

Réponse Q16): $\frac{1}{2}$