

Travail d'algorithmique pour le 23 janvier 2012. Classe de PS1. 7 groupes de 3 élèves et 1 groupe de 4 élèves.
PROGRAMMATION D'UN JEU VIDEO.

Un joueur joue à un jeu vidéo qui consiste à effectuer 100 déplacements de 30 pixels sur une console de jeu dont la résolution (largeur x hauteur) est de 480 pixels sur 360 pixels.

A l'instant 0, on est en O. Le joueur jette un dé tétraédrique dont les faces équilibrées sont numérotées de 1 à 4.

S'il jette un 1, il se déplace de 30 px vers la droite.

S'il jette un 2, il se déplace de 30 px vers la gauche.

S'il jette un 3, il se déplace de 30 px vers le haut.

S'il jette un 4, il se déplace de 30 px vers le bas.

Le jeu se termine :

- dès que l'on doit sortir de l'écran même si les 100 déplacements n'ont pas tous été faits.

- dès que les 100 déplacements sont effectués, mais alors les déplacements du joueur sont tous visibles à l'écran.

A la fin du jeu :

- si on est revenu en O, on gagne 1000 €.

- si on est à 30 px de O ou moins, on gagne 500 €.

- si on est à 60 px de O ou moins, on gagne 200 €.

- si on est à 120 px de O ou moins, on gagne 10 €.

- on perd 2 € sinon.

Le joueur se pose la question suivante : "**Le jeu est-il équitable ?**".

Nous allons tenter de répondre à cette question.

Première partie : Simulation d'un jeu

1. Ecrire un algorithme qui simule ce jeu. (Voir exercice 56 page 310)

Préciser les variables utilisées.

2. Ecrire le programme sur la calculatrice TI82 pour lequel on affichera le codage du déplacement obtenu pour chaque jeu et le gain obtenu par le joueur à la fin du jeu.

Recopier ce programme sur la copie ou imprimer le programme écrit sur le TI Editor (http://www.tiforge.info/ti_editor/).

3. Ecrire le programme AlgoBox pour lequel on affichera le codage du déplacement obtenu pour chaque jeu et le gain obtenu par le joueur à la fin du jeu. On affichera pour contrôle les coordonnées du point obtenu après le dernier déplacement.

4. Ecrire le programme Scratch pour lequel on affichera le codage du déplacement obtenu pour chaque jeu et le gain obtenu par le joueur à la fin du jeu. On affichera pour contrôle les coordonnées du point obtenu après le dernier déplacement.

Deuxième partie : Simulation de N jeux

1. Ecrire un algorithme qui simule N jeux.

Préciser les variables utilisées.

2. Ecrire un programme Scratch qui simule N jeux.

Utiliser un lutin séparé pour calculer le score du jeu.

Utiliser quatre listes pour stocker : 1) le codage des ballades de chaque jeu, 2) les nombres de pas (de 30 px) faits, 3) les gains obtenus, 4) les coordonnées du point obtenu à la fin du dernier déplacement.

Tester votre programme avec N=10 jeux.

Recopier dans un fichier et sur la copie le codage des ballades des 10 jeux obtenus, par exemple :

01. 44114214121142231233234414324431212133124123221142211212144123314131121444

02. 2432123424424341342434121344234311142133324324233131124434342342121313234141232331213313

03. 3113131221333

04. 2134423113223331233

05. 441322333123212112121311442311321134112323

06. 33222411431333224443424344422314142444

07. 33123231413141242413314211233233

08. 3241211232413221332444134211241334324243431232441224143121431313432324224211221343231431313221422314

09. 23333344411422311213222111234123444321422414234314324224343114441323341312333133444323144343133233

10. 3322234231311414231431112411414341412222411112312421232111141

Ainsi que les nombres de déplacements correspondants :

01. 74

02. 89

03. 13

04. 19

05. 42

06. 38
07. 32
08. 100
09. 98
10. 62

Ainsi que les gains obtenus :

01. -2
02. -2
03. -2
04. -2
05. -2
06. -2
07. -2
08. -2
09. -2
10. -2

Et enfin, pour contrôle, les coordonnées du point obtenu à la fin du dernier déplacement.

01. (180,-180)
02. (-30,180)
03. (90,180)
04. (-30,180)
05. (120,180)
06. (-120,-180)
07. (60,180)
08. (-120,60)
09. (0,180)
10. (240,-60)

Troisième partie : Calculs et Espérance

On associe à un jeu la variable aléatoire G qui indique le gain obtenu à la fin de ce jeu.

Sachant que la fréquence tend vers la probabilité lorsque l'on effectue un grand nombre d'expériences aléatoires, effectuer 1000 jeux, stocker les gains et les coller dans une feuille de calcul Libre Office ou Open Office Calc. Calculer dans cette feuille de calcul les fréquences obtenues ainsi que l'espérance de G .

Sur la copie, écrire la loi de probabilité de G et l'espérance $E(G)$.

Finalement, conclure en répondant à la question que le joueur se pose.

Consignes :

On rendra sur clé USB tous les travaux effectués en respectant les consignes suivantes :

- créer un dossier sur votre clé avec le nom du groupe
- mettre tous les documents rendus dans ce dossier en respectant l'arborescence demandée ci-dessous :

/Groupe1

```
|— ElevesDuGroupe.txt
|— Première Partie
|   |— AlgorithmePourUnJeu.txt
|   |— ProgrammeTI82PourUnJeu.txt
|   |— UnJeu.alg
|   |— UnJeu.gif
|   |— UnJeu.sb
|— Deuxième Partie
|   |— 10Coordonnees.txt
|   |— 10Gains.txt
|   |— 10Jeux.txt
|   |— 10Pas.txt
|   |— PlusieursJeuxAvecScoresEtCoordonnees.sb
|   |— ScriptDesJeux.gif
|   |— ScriptDuScore.gif
|— Troisième Partie
|   |— 1000Coordonnees.txt
|   |— 1000Gains.txt
|   |— 1000Jeux.txt
|   |— 1000Pas.txt
|   |— FeuilleCalculEsperance1000Familles.ods
```

Tous les programmes seront transcrits sur papier, soit par recopie (pour le programme calculatrice), soit par impression pour les programmes AlgoBox et Scratch. Pour tous les programmes, on vérifiera soigneusement que les coordonnées du point obtenu à la fin du dernier déplacement ne sortent pas de l'écran de la console de jeu.