

Structures de données et algorithmes fondamentaux

LA TRAVERSÉE DU DÉSERT

Aujourd'hui, nous allons programmer le jeu du chameau, créé par le Heath Users Group et publié dans le magasin More BASIC Computer Games en 1979 :

Dans la version originale du jeu, l'objectif est de voyager 200 miles à travers le grand désert de Gobi. Vous êtes pourchassé(e) par une tribu de natifs. Vous avez une gourde à moitié vide qui peut-être remplie si vous trouvez un oasis ou partiellement remplie, si vous avez la chance d'être aidé par un autre voyageur. Durant le voyage, vous pouvez rencontrer des difficultés comme des tempêtes de sable et votre chameau peut mourir de fatigue. Exemple d'exécution du jeu :

LE JEU DU CHAMEAU !

Vous avez volé un chameau pour traverser le grand désert. Les natifs veulent le récupérer.
Votre objectif est de survivre à la traversée de 300 km sans être attrapé(e).

O P T I O N S :

1. Boire
2. Avancer normalement
3. Avancer à toute vitesse
4. Repos
5. Espérer de l'aide
- T. Terminer la partie

Qu'allez-vous faire ? 2
Vous avez avancé de 11 km.

Vous avez parcouru un total de 11 km jusqu'ici.
Les natifs sont à 19 km derrière vous.
Vous avez soif.
Votre gourde contient 6 gorgées d'eau.
Votre chameau est un peu fatigué.

O P T I O N S :

1. Boire
2. Avancer normalement
3. Avancer à toute vitesse
4. Repos
5. Espérer de l'aide
- T. Terminer la partie

Qu'allez-vous faire ? 2
Vous avez avancé 11 km.

Vous avez parcouru un total de 22 km jusqu'ici.
Les natifs sont à 16 km derrière vous.
Vous avez beaucoup soif !
Votre gourde contient 6 gorgées d'eau.
Votre chameau est fatigué !

O P T I O N S :

1. Boire
2. Avancer normalement
3. Avancer à toute vitesse
4. Repos
5. Espérer de l'aide
- T. Terminer la partie

Qu'allez-vous faire ? 1
Vous avez bu une gorgée d'eau.

Vous avez parcouru un total de 22 km jusqu'ici.
Les natifs vous ont attrapé !

Voulez-vous jouer une nouvelle partie ? non
>>>

Bien sûr, vous êtes libre de changer le thème de votre jeu. Vous pouvez imaginer un voyage à travers le désert de Dune, ou de Tatooine, ou à travers la galaxie, ou la forêt de Fangorn, ou bien, vous pouvez imaginer que des pirates vous pourchassent dans l'océan. Vous pouvez remplacer le chameau par un vaisseau spatial ou un bateau, ou un autre type de transport. La fatigue du chameau peut-être remplacée par le combustible du vaisseau et les tempêtes de sable peuvent être remplacés par des ouragans, des voleurs ou des monstres, etc.

1 Le jeu de base

1. Déclarer les constantes qui serviront à paramétrer le jeu :

```
KM_VOYAGE = 300      # Distance à parcourir pour gagner.
KM_NORM_MIN = 10     # Distance min. à la vitesse normale.
KM_NORM_MAX = 15     # Distance max. à la vitesse normale.
KM_RAP_MIN = 20      # Distance min. à toute vitesse.
KM_RAP_MAX = 25      # Distance max. à toute vitesse.
AVANTAGE_VOYAGEUR = 20 # L'avantage initial du voyageur.
GOURDE_PLEINE = 12   # Le nombre de gorgées de la gourde.
MORT_SOIF = 4        # Nombre de tours pour mourir de soif.
MORT_FATIGUE = 4     # Nombre de tours pour mourir de fatigue.
DIF_AIDE = 3         # Difficulté pour trouver de l'aide.
AVANCE_NATIFS = 4    # Vitesse des natifs.
```

2. Ajouter les instructions qu'affichent l'introduction du jeu. Utiliser une instruction `print` par ligne.

```
LE JEU DU CHAMEAU !

Vous avez volé un chameau pour traverser le grand désert.
Les natifs veulent le récupérer.
Votre objectif est de survivre à la traversée de 300 km sans être attrapé(e).
```

3. Ajouter l'initialisation des variables de contrôle du jeu :

```
km_voyageur = 0      # Distance totale parcourue.
km_natifs = km_voyageur - AVANTAGE_VOYAGEUR # Distance parcourue par les natifs.
gourde = GOURDE_PLEINE // 2 # Nombre de gorgées dans la gourde.
soif = 0             # Niveau de soif du voyageur.
fatigue = 0          # Niveau de fatigue du chameau.
```

4. Créer la boucle principale du jeu. Utiliser une instruction `while` pour afficher les options tant que l'utilisateur n'a pas choisi de terminer la partie.

```
O P T I O N S :
1. Boire
2. Avancer normalement
3. Avancer à toute vitesse
4. Repos
5. Espérer de l'aide
T. Terminer la partie

Qu'allez-vous faire ?
```

5. Ajouter une instruction `if-elif-else` pour vérifier l'option choisie par le joueur et changer les variables de contrôle correspondantes. Pour faciliter la programmation, faites les ajouts un par un et testez les résultats avant de passer au prochain cas :
- Si le joueur choisit « Boire », vérifier si la gourde est vide (`gourde == 0`).
 - Dans ce cas, afficher « La gourde est vide! ».
 - Sinon, affecter zéro à la variable `soif` pour indiquer que le voyageur n'a plus soif. Soustraire également une gorgée à la gourde. Afficher « Vous avez bu une gorgée. ».
 - Si le joueur choisit « Avancer normalement » :
 - Ajouter 1 au niveau de fatigue du chameau.
 - Choisir aléatoirement un nombre entre `KM_NORM_MIN` et `KM_NORM_MAX` qui correspondra à la distance parcourue par le voyageur.
 - Ajouter la distance parcourue à `km_voyageur`.
 - Afficher la distance parcourue.
 - Si le joueur a choisi « Avancer à toute vitesse » :
 - Ajouter 2 au niveau de fatigue du chameau.
 - Choisir aléatoirement un nombre entre `KM_RAP_MIN` et `KM_RAP_MAX` qui correspondra à la distance parcourue par le voyageur.
 - Ajouter la distance parcourue à `km_voyageur`.
 - Afficher la distance parcourue.
 - Si le joueur choisit « Repos », affecter zéro à la variable `fatigue` pour indiquer que le chameau s'est reposé. Afficher « Votre chameau s'est bien reposé. ».
 - Si le joueur choisit « Espérer de l'aide » :
 - Choisir aléatoirement un nombre entre 0 et `DIF_AIDE` pour décider si le joueur a rencontré quelqu'un sur son chemin.
 - Si le nombre choisi est 0 :
 - Afficher « Vous avez trouvé de l'aide ».
 - Vérifier si la gourde est pleine (`gourde >= GOURDE_PLEINE`) :
 - Si la gourde est pleine, afficher « La gourde est déjà pleine! ».
 - Sinon, ajouter au plus 3 à la variable `gourde`. (Attention : le niveau de la gourde ne doit pas dépasser `GOURDE_PLEINE`). Afficher « Quelques gorgées ont été ajoutées à votre gourde. ».
 - Sinon, afficher « Vous n'avez pas trouvé de l'aide. ».
 - Si le joueur ne choisit aucune des options du menu, afficher « Option invalide! » et redemander l'action du joueur.
6. Quand le joueur choisit une option valide, vous devez traiter aussi les autres événements du jeu. Ajouter du code pour traiter les événements ci-dessous. Pour faciliter la programmation, ajouter les items un par un et testez le résultat avant de passer au prochain cas :

Avancement du voyageur :

- Si la distance totale parcourue par le voyageur excède `KM_VOYAGE`, félicitez le joueur et terminez le jeu. Le joueur a gagné!!
- Sinon, affichez la distance totale parcourue jusqu'ici.

Avancement des natifs :

- Choisir aléatoirement un nombre entre 0 et `AVANCE_NATIFS` pour l'avancement des natifs.
- Si le nombre choisi est 0, alors les natifs avancent à toute vitesse. Ceci est fait de la même manière que pour le voyageur.
- Si le nombre choisi est 1, alors les natifs avancent à une vitesse normale. Ceci est fait de la même manière que pour le voyageur.
- Si les natifs ont attrapé le voyageur (`km_natif >= km_voyageur`), afficher « Vous avez été attrapé par les natifs! » et terminer le jeu. Le joueur a perdu!!

- Sinon, afficher l'avantage que le voyageur a sur les natifs.

Soif :

- Ajouter 1 à la soif du voyageur.
- Si la soif dépasse `MORT_SOIF`, alors afficher que le voyageur est mort de soif et terminer le jeu. Le joueur a perdu!!
- Sinon, selon le niveau de soif du voyageur, afficher :
 - « Vous n'avez pas soif. »
 - « Vous avez un peu soif. »
 - « Vous avez beaucoup soif! »
 - « Vous allez mourir de soif!! »

Gourde : Afficher le nombre de gorgées dans la gourde.

Fatigue du chameau :

- Si la fatigue du chameau dépasse `MORT_FATIGUE`, alors afficher que le chameau est mort de fatigue. Afficher également que les natifs ont attrapé le voyageur. Le joueur a perdu!!
- Sinon, selon le niveau de fatigue du chameau, afficher :
 - « Le chameau est en bonne forme. »
 - « Le chameau est un peu fatigué. »
 - « Le chameau est très fatigué! »
 - « Le chameau va mourir de fatigue!! »

7. Ajouter une boucle pour que l'utilisateur puisse jouer plusieurs parties sans redémarrer le programme. À la fin de chaque partie, demander « Voulez-vous jouer une nouvelle partie? ».

8. Tester votre jeu pour être sûr qu'il n'y a pas de bugs...

Il y a beaucoup d'éléments aléatoires dans ce jeu. Même avec une bonne stratégie, il est très difficile de gagner. Si vous le souhaitez, vous pouvez changer les paramètres (constantes) pour le rendre plus facile.

2 L'extension : les éléments spéciaux.

Rajouter deux éléments spéciaux dans le jeu. À chaque tour du jeu, choisir aléatoirement un nombre entre 0 et `SPECIAL` (nouveau paramètre de jeu à ajouter). Si le nombre choisi est 0, le joueur a trouvé un oasis. Si le nombre choisi est 1, une tempête de sable s'abat sur le désert.

2.1 Le voyageur trouve un oasis

Dans ce cas, le joueur peut remplir complètement sa gourde au passage. Le joueur ne perd pas de temps : les natifs n'avancent pas pendant cette action du joueur.

2.2 Une tempête de sable s'abat sur le désert

Dans ce cas, le joueur ne peut rien faire, sauf attendre. À cause de la violence de la tempête, le voyageur n'arrive pas à calmer son chameau. Le joueur ne peut même pas boire ni tenter de trouver de l'aide. Le chameau est également trop stressé pour se reposer. Les natifs, par contre, sont habitués et avancent normalement.

3 À vous de jouer (bonus)!!

Imaginez d'autres éléments et difficultés pour votre jeu.

Quelques exemples :

- Choisir aléatoirement le nombre de tours que durera la tempête.
- Proposez au voyageur de se reposer et/ou boire pendant les tempêtes de sable.
- Le voyageur se perd dans le désert. La traversée sera de X km (tiré au sort) plus longue!
- Une petite tempête de sable permet au voyageur d'avancer quand même, mais plus lentement. La distance parcourue sera moindre et le niveau de fatigue du chameau sera plus élevé.
- Les possibilités sont infinies . . .