

TheBigAdventure : Manuel utilisateur

Sommaire

I. Préambule.....	2
II. Lancement et ligne de commande.....	2
Lancement du jeu.....	2
Options du jeu.....	2
III. Le jeu.....	3
La carte.....	3
Actions du joueur.....	4
Objectif du jeu.....	5
IV. Création/édition des cartes.....	5
Grid.....	6
Element.....	7
Apparences associées aux types de tuiles.....	8
Conseils et avertissements.....	8

I. Préambule

Le jeu auquel vous allez jouer est un jeu d'aventure où l'on peut créer/éditer ses propres cartes d'aventure. Le principe du jeu se rapproche plus d'un jeu où l'on crée sa propre histoire avec des objets à trouver et des ennemis à tuer .

Il se base sur la bibliothèque Zen5 et sur les tuiles du jeu « Baba Is You ».

Le jeu se joue exclusivement au clavier avec les touches de directions, I et ESPACE.

Deux cartes sont déjà disponibles dans le répertoire maps : demo.map (une carte de démo du jeu) et adventure.map (une grande carte d'aventure).

II. Lancement et ligne de commande

Lancement du jeu

Le jeu se lance à l'aide du fichier jar présent dans l'archive fournie, il suffit alors d'utiliser la commande : `java -jar thebigadventure.jar [OPTIONS]` et le jeu se lance.

Options du jeu

Le programme comporte plusieurs options à ajouter à la ligne de commande avant le lancement du jeu :

→ **--level** *name.map* : Option à mettre **obligatoirement** pour jouer, elle permet de valider et charger une carte du jeu (un fichier .map). Si la carte n'est pas valide une erreur sera signalée et le jeu ne se lancera pas. Pour spécifier la carte il faut mettre son chemin (ex : maps/adventure.map).

→ **--validate** : Option à mettre si vous souhaitez uniquement valider la carte. Si elle est présente, un message de validation sera affiché (OK si tout va bien ou une erreur si la carte comporte une erreur) mais le jeu ne sera pas lancé.

→ **--dry-run** : Si cette option est active, les ennemis de la carte ne bougeront pas. Attention car il peuvent toujours vous attaquer dans ce mode !

III. Le jeu

Une fois la ligne de commande entrée, le jeu prend 2 à 4 secondes à se lancer et s'afficher. Une fois fait, la fenêtre apparaît avec la carte que vous avez choisi de jouer.



Figure 1: Le jeu avec la carte *adventure.map* chargée

La carte

La carte se compose de plusieurs éléments :

Le joueur (vous), ici le personnage nommé « fluffy » que vous pouvez contrôler

Les ennemis (les méchants), ici les personnages nommés « LevelOne ». Ils peuvent s'attaquer à vous si vous êtes devant eux et qu'ils vous font face, cependant ils peuvent rater leurs attaques. En fait ils ne sont pas très bons pour manier l'art de l'attaque (contrairement à vous) ce qui fait qu'ils n'ont que 20 % de chances de vous attaquer. Attention à vous car de nombreuses attaques répétées peuvent avoir raison de vous !

Les objets (à ramasser), ici le bâton en haut à droite. Il en existe plusieurs types, les armes qui font des dégâts aux ennemis, la nourriture qui vous rend 25 % de vos points de vie maximaux et les objets échangeables (le reste) qui auraient pu être échangeables mais ne le sont pas. Passer sur un objet le récupère dans votre inventaire.

Les obstacles, ici les arbres, que ni vous ou les ennemis ne peuvent traverser.

Les décorations, ici les fleurs ou buissons, que vous pouvez traverser

Actions du joueur

En tant que joueur et ayant le pouvoir du clavier, vous avez à votre disposition plusieurs actions possibles dans le jeu :

- se déplacer : avec les flèches du claviers, pour aller à gauche, droite, haut ou bas. Cependant vous ne pouvez pas traverser les obstacles ou les ennemis comme dit précédemment.
- ouvrir l'inventaire : avec la touche i. Une fois dans l'inventaire, vous pouvez naviguer parmi les objets avec les flèches du clavier, vous pouvez sélectionner un objets à mettre dans votre main avec la barre espace et sinon vous pouvez le quitter sans sélectionner un objet avec i. Un objet dans l'inventaire a un nom et une quantité indiqués sur le côté.
- utiliser un objet : avec la barre espace. Cette action dépend de l'objet tenu en main, si c'est une arme et que vous faites face à l'ennemi (il doit être dans la direction où vous frapper) vous lui infligerez des dégâts. Si c'est de la nourriture, vous la consommerez et vous gagnerez 30 % de votre vie initiale. Les autres objets ne font rien malheureusement.
- quitter le jeu : avec la touche q. Vous pourrez quitter le jeu à tout moment avec la touche q que vous soyez dans l'inventaire ou sur la carte.

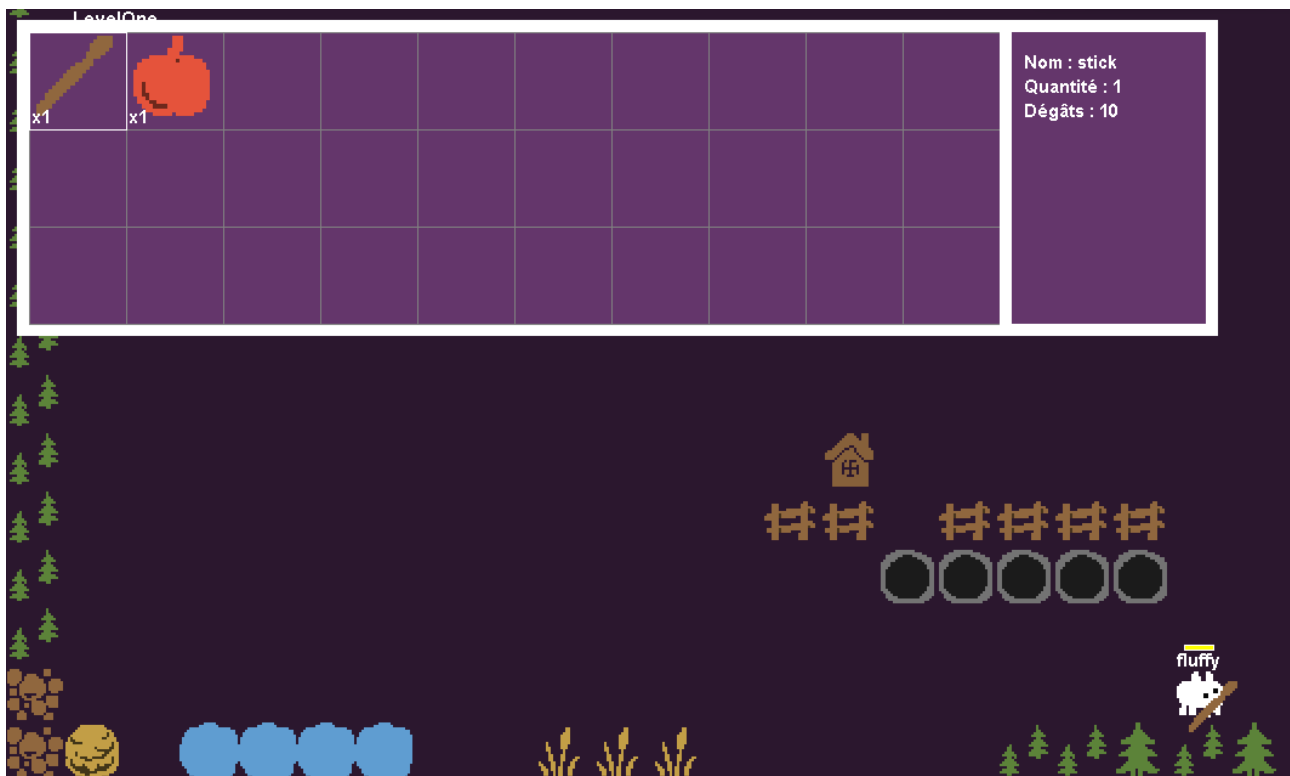


Figure 2: Inventaire ouvert avec l'objet stick sélectionné



Figure 3: Le joueur faisant face à un ennemi pour le frapper



Figure 4: Le joueur avec un bâton en main, la barre jaune est la barre de vie

Objectif du jeu

Dans chaque partie ou carte que vous jouerez votre objectif principal sera de tuer tous les ennemis de la carte sans mourir. Si jamais vous mourez, la partie se termine et vous devrez recommencer. Il faut donc chercher dans la carte si des armes puissantes ne sont pas dissimulées !

Quand la partie est terminée, un écran de fin apparaît et vous devrez appuyer sur n'importe quelle touche pour fermer le jeu.

IV. Création/édition des cartes

Savoir comment jouer c'est une chose mais savoir comment créer les cartes c'en est une autre ! Les cartes sont chargées à partir d'un fichier en .map similaire à un fichier texte mais avec un format à respecter.

Ce fichier s'organise en 2 types de données [grid] et [element]. Le 1^{er} permet de définir la grille et ses éléments répétitifs, par exemple les décorations, les obstacles, les dimensions etc. Le 2^{ème} permet de définir les éléments « uniques » de la carte comme le joueurs, les ennemis, les objets ... et aussi des obstacles ou décorations uniques.

Grid

Les données de type [grid] sont définies de cette manière :

- size : qui fixe la taille en hauteur et largeur de la grille
- encodings : qui permet d'associer une lettre à une apparence de tuile
- data : qui permet de placer les éléments définis dans encodings

Exemple :

```

[grid]
size: (20 x 25)
encodings: GRASS(g) TREE(T) WALL(W)
data: ""
TTTTTTTTTTTTTTTTTTTT
T                    T
T                    T
T      W            T
T                    T
T                    T
T                    T
T      gg           T
T    gggggggg       T
T      gggg         T
T    gg  g          T
T                    T
T                    T
T                    T
T                    T
T                    T
T                    T
T                    T
T                    T
T                    T
T                    T
TTTTTTTTTTTTTTTTTTTT
""
  
```

Figure 5: Section [grid] de la carte demo.map

Element

Les données de type [element] sont définies de cette manière : (ici tous les champs ne sont pas obligatoires selon la donnée définie)

- name: le nom de l'élément
- skin : apparence (plus de précision dans la section suivante)
- player : true si l'élément est le joueur (un seul possible → le 1^{er} défini dans le fichier)
- position : position de départ sur la carte (0, 0) est le coin en haut à gauche
- health : les points de vie (entier positif)
- kind : type d'élément parmi enemy, item et obstacles
- zone : zone définie par une coordonnée et une taille dans laquelle se balade un ennemi
- behavior : comportement de l'ennemi (stroll possible uniquement)
- damage : dégât infligé à chaque attaque (entier positif)

Exemple :

```
[element]
name: fluffy
player: true
skin: BABA
position: (10,10)
health: 100

[element]
name: durenale
skin: SWORD
position: (2,2)
kind: item
damage: 15

[element]
name: stick
skin: STICK
position: (4,2)
kind: item
damage: 5

[element]
name: key
skin: KEY
position: (6,2)
kind: item

[element]
name: waldo
skin: CRAB
position: (8, 2)
kind: enemy
health: 50
zone: (8, 2) (3 x 3)
behavior: stroll
damage: 10

[element]
name: piano
skin: PIANO
position: (18,10)
kind: obstacle
```

Figure 6: Définition de plusieurs éléments du fichier demo.map

Apparences associées aux types de tuiles

Si la plupart des informations des éléments sont données dans les sous sections de [element], certains éléments vont prendre des types différents selon leurs skin :

Décorations → ALGAE, CLOUD, BUBBLE, DUST, FLOWER, FOLIAGE, GRASS, LADDER, LILY, PLANK, REED, ROAD, SPROUT, TILE, TRACK, VINE, WIND

Obstacles → BED, BOG, BOMB, BRICK, CHAIR, CLIFF, DOOR, FENCE, FORT, GATE, HEDGE, HOUSE, HUSK, HUSKS, ICE, LOCK, LAVA, MONITOR, PIANO, PILLAR, PIPE, ROCK, RUBBLE, SHELL, SIGN, SPIKE, STATUE, STUMP, TABLE, TOWER, TREE, TREES, WALL, WATER

Enemis → BABA, BADBAD, BAT, BEE, BIRD, BUG, BUNNY, CAT, CRAB, DOG, FISH, FOFO, FROG, GHOST, IT, JELLY, JIJ, KEKE, LIZARD, ME, MONSTER, ROBOT, SNAIL, SKULL, TEETH, TURTLE, WORM

Joueur → BABA, BADBAD, FOFO, IT

Armes → SWORD, STICK

Nourriture → BANANA, BOBA, BOTTLE, BURGER, CAKE, CHEESE, DONUT, DRINK, EGG, FRUIT, FUNGUS, FUNGI, LOVE, PIZZA, POTATO, PUMPKIN, TURNIP

Autres objets → BOOK, BOLT, BOX, CASH, CLOCK, COG, CRYSTAL, CUP, DRUM, FLAG, GEM, GUITAR, HIHAT, KEY, LAMP, LEAF, MIRROR, MOON, ORB, PANTS, PAPER, PLANET, RING, ROSE, SAX, SCISSORS, SEED, SHIRT, SHOVEL, STAR, STICK, SUN, SWORD, TRUMPET, VASE

Conseils et avertissements

La section [grid] doit toujours être définie dans l'ordre : size, encodings, data !

Pour ce qui est des [element], leurs sous sections peuvent être définies dans n'importe quel ordre et un [element] peut être défini avant [grid] et inversement

Nous vous conseillons toutefois de suivre le modèle donné dans demo.map et adventure.map pour que votre fichier soit mieux organisé mais vous pouvez faire comme vous le souhaitez.

Prenez bien en compte le temps d'attaque des ennemis pour pouvoir bien équilibrer vos cartes (ou complètement les casser si vous voulez pousser à l'extrême le jeu).

Si jamais vous faites des erreurs, le programme vous prévient. Cependant, il ne vous dit pas si vous avez mis le joueur sur un obstacle par exemple donc faites attention.

Ce manuel touche à sa fin, nous vous souhaitons donc une bonne partie et bonne chance pour la création de vos cartes !