Baba is You

PROJET

Guide utilisateur

RAMAROSON Johan
Groupe 3

GUYOLLOT Alan
Groupe 3

Du 28 Octobre au 16 Janvier 2021

Chargé de TD : Monsieur Forax
Matière : Java Programmation Orienté Objet

Etablissement / Formation :
Université Gustave Eiffel – Licence 3 Informatique Semestre 5

Introduction

Objectif général

Le but de ce projet est de développer un jeu du type « Baba Is You » qui est un jeu où chaque niveau est une énigme qu'il nous faut résoudre en modifiant les règles de l'environnement.
Pour se faire, vous trouverez, dans chaque niveau, des phrases composées de mots (exclusivement en anglais) qui s'avèrent être des blocs déplaçable par le joueur. Chacun de ses mots va définir les propriétés de l'environnement.

Condition de rendu

Ce projet est à faire par binôme (2 personnes, ni 1 ni 3) et doit être déposé au plus tard le 16 janvier 2021 à 23h59 sur e-learning.

Une soutenance « bêta » aura lieu avant les vacances de noël 2020 (la date et l'horaire vous seront précisés). Votre travail sera noté mais vous pourrez tenir compte des remarques pour améliorer votre projet jusqu'au rendu final.

Règles du jeu

Dans ce jeu, les règles sont affichés à l'écran en tant que blocs de mots.
Le concept du jeu est : ***changer les règles du jeu pour gagner !***

Vous pouvez absolument tout changer ! Que ce soit le personnage que vous contrôler ou alors les propriétés un objet de l'écran.

Il existe 3 types de blocs de mots :

- les **noms (Nouns)** **:** un objet avec lequel interagir (représenté dans le jeu par une image)
- les **opérateurs (Operators)** **:** Dans notre jeu, nous avons que ***IS*** (est) qui va affecter des propriétés/noms à un nom
- les **propriétés (Properties)** **:** une propriété qu'on va attribuer (ou pas selon l'opérateur) aux noms

D'accord mais comment écrire des règles ?
Pour écrire une règle, il faut garder en tête que :

Les règles ont une orientation, ...
Il y a deux façons d'écrire des règles :

- ✔ Horizontalement
- ✔ Verticalement

- ✗ **Pas en diagonale**

un sens, ...
Les règles se lisent exclusivement de :

- gauche à droite
- de haut en bas

et un ordre
En effet, on ne peut pas mettre les blocs de mots n'importe comment.

On peut mettre dans les ordres (en suivant le sens de lecture d'une règle):

- Un **nom** suivi d'un **opérateur** puis d'une **propriété**

Les règles dans cet ordre vont attribuer la propriété au nom selon l'opérateur (On n'a que ***IS*** comme opérateur donc ici la propriété sera donné au nom)

Exemple : ***Rock IS STOP***
Ici, on attribue la propriété STOP aux rochers ce qui signifie que le joueur ne pourra pas pousser les rochers.

- Un **nom1** suivi d'un **opérateur** puis d'un **nom2**

Les règles dans cet ordre vont modifier tous les nom1 en nom2 selon l'opérateur (On n'a que ***IS*** comme opérateur donc ici tous les nom1 vont devenir des nom2)

Exemple : ***Wall IS Flag***
Avec cette règle, tous les murs **(Wall)** du jeu vont devenir des drapeaux **(Flag)**

Pour voir tous les noms, opérateurs et propriétés disponibles, vous pouvez aller voir l'annexe 1 (pages 8 à 10)

Pour illustrer tout ça, prenons quelques exemples :

Exemple 1 : La possibilité de pousser des pierres
Pour avoir cette règle, nous devons avoir nos blocs de mots disposés de la manière suivante :

OU

Rock désigne les pierres
Is désigne l'opérateur qu'on va utiliser entre deux textes
Push désigne la propriété qu'on va affecter au nom (Ici la propriété pour pousser)

Remarque : Vous pouvez utiliser le **même opérateur** pour faire deux règles (une horizontale et une verticale).

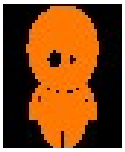
Exemples 2 et 3 : Le drapeau permet de gagner et les rochers se transforment en murs

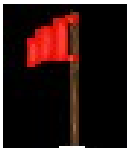
Rock désigne les pierres, ***Wall*** les murs et ***Flag*** les drapeaux
Is désigne l'opérateur qu'on va utiliser entre les textes
Win désigne la propriété qu'on va affecter au nom (la propriété pour gagner dans notre cas)

Annexe 1

Listes des blocs de mots disponibles

Pour les noms:

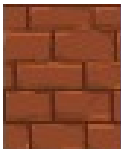
Baba 

Flag 

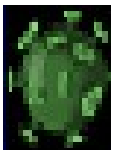
Lava 

Rock 

Skull 

Wall 

Water 

Virus 

Pour les opérateurs :

IS 

Pour les propriétés (associé à un nom qu'on appelera ici objet) :

Push : Cet objet (le nom qui a la propriété PUSH) est solide et peut être poussé en se déplaçant vers lui lorsqu'il est adjacent à lui. S'il y a plusieurs éléments **PUSH** d'affilée, tous seront poussés.

Sink : Si un objet chevauche sur cet objet(ayant **Sink)**, cet objet est supprimé, ainsi que tous les objets (non flottants) qui le chevauchent.

Defeat : Si un **objet YOU** chevauche cet objet, **l'objet YOU** est supprimé.

You : Le joueur peut contrôler cet objet à l'aide des commandes de mouvement (ici les touches directionnelles)

Stop : Cet objet est solide.

Win : Si un **objet YOU** (un nom ayant la propriété **YOU**) chevauche cet objet **(ayant Win)**, le niveau est gagné.

Melt : Si cet objet chevauche un **objet HOT,** cet objet est supprimé.

Hot : Si un **objet MELT** chevauche cet objet, l'objet **MELT** est supprimé.

Immune : Si un **objet YOU** (un nom ayant la propriété You) a la propriété **IMMUNE** et chevauche un objet **DEFEAT**, alors le joueur ne perd pas

Spread : Cet objet se propage en se multipliant sur le jeu

Annexe 2

Ordre d’application des règles

A chaque tour, les choses suivantes se produisent dans cet ordre :

- Les mouvements du joueur
- Les directions se mettent à jour
- PUSH est vérifié après chaque mouvement

La priorité des propriétés est la suivante :

- Push > Hot
- Stop > Push
- Vaccin > Defeat

Annexe 3

Commandes dans le terminal

appuyez sur Entrée après chaque ligne verte ou rouge
Après avoir fait un

ant jar

Le .jar sera généré -> ***baba.jar (à faire uniquement une fois au tout début)***

Pour exécuter le programme il faut faire :

java --enable-preview -jar baba.jar [paramètres optionnels]

A mettre à la place de [paramètres optionnels]
Il est possible de paramétrer le jeu en ligne de commande à l'aide des options suivantes :

- ***--levels name***: nom du dossier contenant les énigmes à résoudre, auquel cas, à chaque niveau terminé on enchaîne sur le niveau suivant;
- ***--level name***: nom du fichier contenant l'énigme à résoudre ;
- ***--execute mot1 mot2 mot3***: permet d'exécuter une phrase comme "ROCK IS FLAG", il peut y avoir plusieurs --execute pour une même ligne de commande (chut, c'est pour tricher);

Exemple : *Supposons qu'on écrive ça dans le terminal (**après avoir écrit*** **ant jar)**

java --enable-preview -jar baba.jar --levels name: listLevel

Cela vous fera faire tous les niveaux du fichiers tant que vous gagnez

---

java --enable-preview -jar baba.jar --level name: level1.txt

Cela lancera le niveau 1

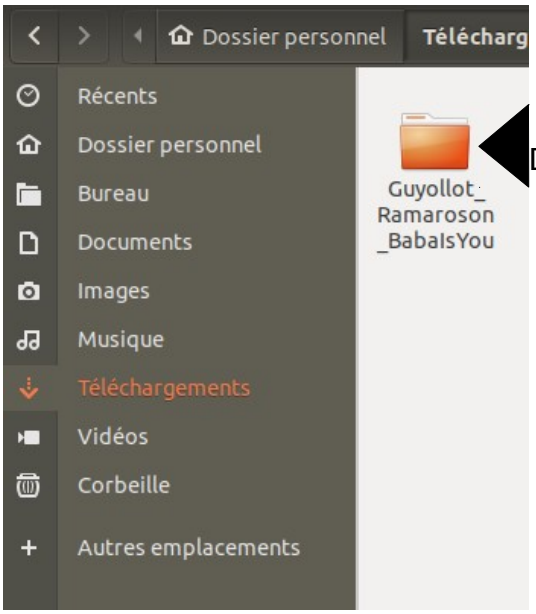
Les phrases vertes sont celles qu'on écrit dans le terminal !

Annexe 4

Guide utilisateur

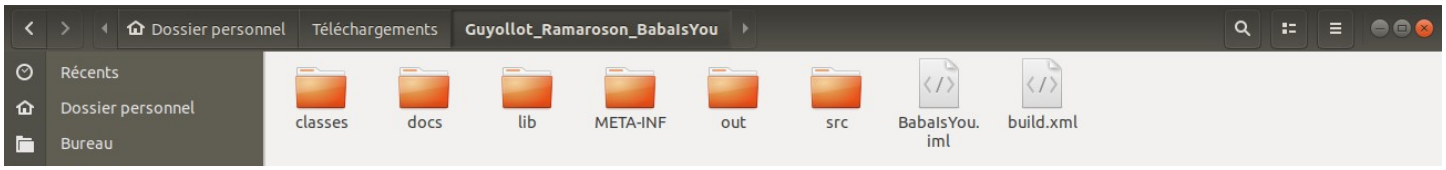
Pour tester le jeu, vous devez télécharger les fichiers du projet et compiler le programme.

Mais comment faire ?
Pour cela, il suffit de télécharger le dossier sur E-learning et d'aller dans vos **Téléchargements.**



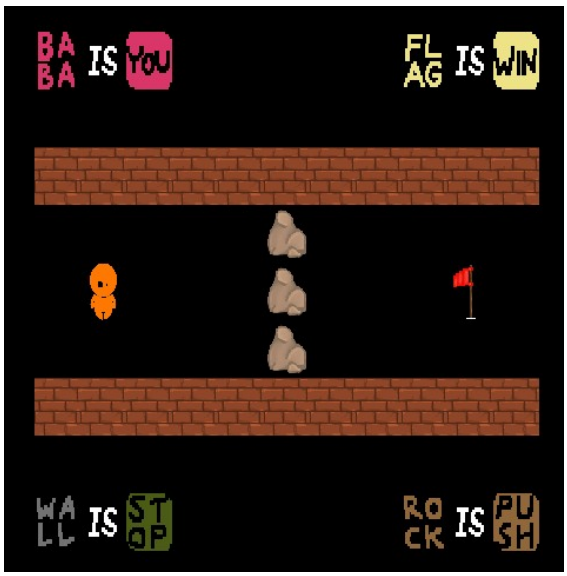
Double cliquez ici

Une fois arrivé ici, faites un clic droit dans le dossier et appuyez sur ouvrir dans le terminal



Par exemple, on veut tester tous les niveaux (va lancer le niveau suivant tant qu'on gagne), on va écrire la commande de l'annexe 3 dans le terminal (celle en rouge) puis appuyer sur **Entrée**. Ça va lancer le programme (ici le niveau 1)

```
igm@deulin:~/Téléchargements/final/Guyollot_Ramaroson_BabaIsYou$ ant jar;
java --enable-preview -jar baba.jar --levels name: listLevel
```



Pour déplacer le joueur (ici Baba), on utilise les flèches directionnelles ou encore les touches Z,D,S et Q pour respectivement HAUT, DROITE, BAS et GAUCHE.

Vous pouvez aussi quitter le programme en appuyant sur la touche **Entrée** ou **M**.

Exemple : On veut faire le niveau 7 avec la règle supplémentaire Baba is MELT
Vous devez tout d'abord écrire :

ant jar (si vous l'avez pas déjà fait)
puis
java --enable-preview -jar baba.jar --level name: level7.txt --execute BABA IS MELT

Cela lancera le niveau 7 avec la nouvelle règle qu'on a directement ajouté dans la commande.