



# Jeu de portes

TROUVE LA SORTIE

M.DRIDI

Moussa Smain | Développement d'application | 28-12-2020

Vandenboren Simon

Pépin Maëlic

## SOMMAIRE

### Développement d'application

<b>SOMMAIRE</b> .....	1
D'où nous est venu cette idée ? .....	2
Description du jeu .....	3
Structures de données.....	5
Fonctionnalités développées .....	5
Organisation du travail .....	7
Conclusion sur l'état du travail.....	7
Video présentation .....	8
Bibliographie / webographie .....	8



## D'où nous est venu cette idée ?

Pour commencer nous avons voulu faire un jeu simple et compréhensible pour qu'il puisse convenir à toutes sortes de personnes. C'est-à-dire qu'il comporte plusieurs niveaux de type facile, difficile mais aussi un mode multijoueur pour plus de fun !

C'est là que nous est venu l'idée d'un jeu suivant la rng c'est-à-dire où l'on va devoir ouvrir des portes aléatoires mais pour rendre ce jeu un peu plus attractant nous avons décidé d'y intégrer de la culture telle que des questions de mathématique pour le rendre plus marrant à jouer et qu'il soit bénéfique à tout le monde.





## DESCRIPTION DU JEU

Les règles pour un joueur :

### MODE FACILE

- Une partie comporte 5 paliers dont 5 portes par palier.
- Seule une porte permet de passer à la suite.
- Sous chaque bonne porte se glisse une question.
- Ouvrir la bonne porte en répondant par la bonne réponse vous permet de passer au palier suivant.
- Si le joueur se trompe de portes ou de réponses, il recommence au premier palier.
- Quand on recommence les portes ne change pas, seules les questions changent.
- On gagne une partie si ont réussi à finir le parcours sans avoir beaucoup de pénalités.

Les règles pour deux joueurs :



- Une partie comporte 5 paliers dont 5 portes par palier.
- Seule une porte permet de passer à la suite.
- Sous chaque bonne porte se glisse une question.
- Ouvrir la bonne porte en répondant par la bonne réponse vous permet de passer au palier suivant.
- Si le joueur se trompe de portes ou de réponses, il recommence au premier palier.
- Quand on recommence les portes ne change pas, seules les questions changent.
- Une partie est remporté si l'on fait moins de temps que son adversaire en ayant bien évidemment moins de pénalités.



## MODE DIFFICILE

Bien sûr pour le mode difficile qui dit mode difficile dit questions plus dures mais aussi on dispose de plusieurs vie ♥ , en cas de plusieurs erreurs à la suite on risque de recommencer à 0 en ayant réinitialisées les portes aléatoires ce qui rend la partie extrêmement difficile voire même impossible.

# *À toi de jouer !*



## STRUCTURES DE DONNEES

Pour ce jeu nous avons utilisés qu'une seule structure qui nous convient amplement.

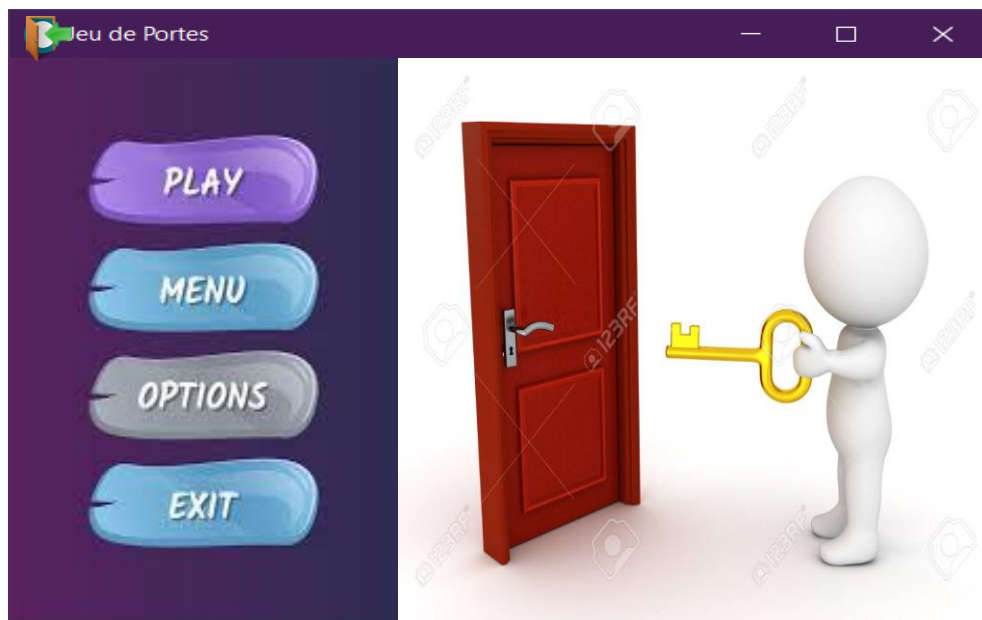
- Notre structure Joueur va comprendre :
  - Le nom du joueur avec lequel on va pouvoir identifier quelle personne est en train de jouer mais aussi pour pouvoir sauvegarder son temps de jeu.
  - Le temps de la partie avec lequel on va pouvoir comparer les scores des joueurs pour pouvoir définir un gagnant mais aussi un score à battre.
  - Les pénalités qu'il a eu durant la partie, ici en seconde suivant le nombre d'erreurs qu'il a fait (qui contient aussi les vies ♥ du joueur)

Avec cette structure on va pouvoir initialiser les joueurs, afficher les joueurs et pouvoir choisir lequel.

## FONCTIONNALITES DEVELOPPEES

Pour l'interface qu'on a pu faire, on a utilisé la SDL et ces librairies pour pouvoir faire une fenêtre uniquement (manque de temps et pas de cours sur la SDL).

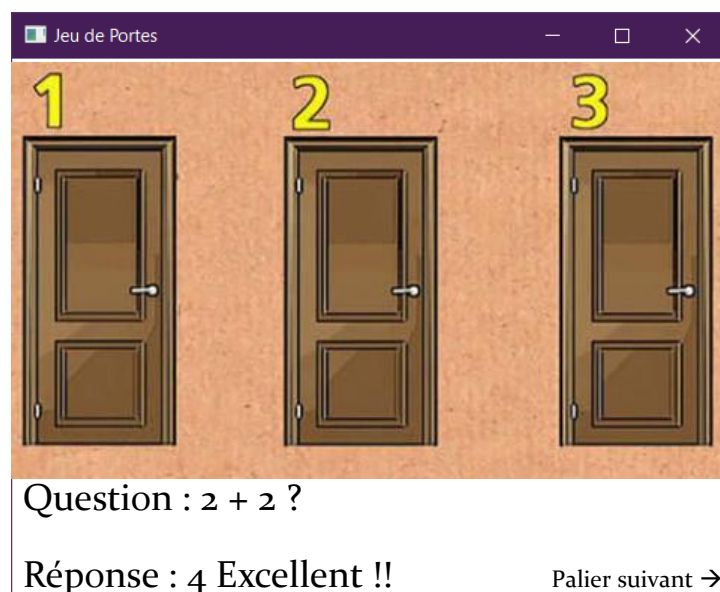
Mais notre projet final aurait ressemblé à ça.



Pour l'interface d'une partie elle aurait eu comme vue.



Si on trouve la bonne porte une question s'affiche et on y répond.



- En fin de partie du premier joueur, on affiche « nomJoueur doit battre ...secondes ».

## ORGANISATION DU TRAVAIL

Nous avons tous les trois utilisés un système d'exploitation qui est Windows, avec GitHub qui nous a beaucoup servi :

<http://github.com/SmainMoussa/Dossier-du-jeu>

Pour la répartition du travail, nous nous sommes fait des partages d'écran via Discord, puis nous avons programmé ensemble :

- Simon a partagé son écran et a travaillé sur les fichiers .c, les lectures de fichiers avec l'aide de Smaïn et Maëlic.
- Smaïn a partagé son écran et a travaillé sur les programmes .c et sur les fichiers .h avec le Makefiles puis l'interface SDL avec l'aide de Maëlic et Simon.
- Maëlic a partagé son écran et a travaillé sur les programmes de lectures question, réponses avec l'aide de Simon et Smaïn.

Puis nous avons fini avec le Readme qui contient les instructions à suivre et ce PDF.

## CONCLUSION SUR L'ETAT DU TRAVAIL

Pour conclure sur où nous en sommes, nous avons un jeu fonctionnel :

- Un menu
- Un mode solo :
  - Facile
  - Difficile
- Un mode multijoueur avec décompte (3, 2, 1)
- Sauvegarde des scores et des noms
- Pour la SDL on n'a pas le temps de suivre des cours sur internet mais on a quand même réussi à faire une interface avec le titre du jeu exécutable.
- Néanmoins nous avons ajouté des fichiers .c et .h qu'on n'a pas encore assemblé avec le reste du jeu.

Pour le mode difficile et facile nous avons rencontré un problème concernant la lecture de fichier pour pouvoir comparer des chaînes de caractères en ouvrant deux fichiers différents. (Résolu)



