

Projet Final – Labyrinthe du Tank

Introduction à la programmation Web client – IFM25909

Introduction

Le but de ce projet est de développer un petit jeu de style arcade que vous pourrez jouer avec vos amis dans le but d’avoir le meilleur score possible. Le jeu sera joué dans votre navigateur web. Fidèle au jeu d’arcade du temps, chaque partie du jeu sera courte et le joueur devra essayer d’amasser le plus de points possibles avant la fin.

Ce document présente l’énoncé de la création de base du projet. Toutefois, l’ajout de fonctionnalités supplémentaire est encouragé. Des points supplémentaires pourront même être donnés dépendant de la fonctionnalité.

Équipe

Le travail se fait seul. Chaque étudiant doit remettre son propre projet. Je veux savoir si chacun d’entre vous comprends bien la matière individuellement.

Tout partage de code entre étudiant sera considéré comme une fraude scolaire. Vous devez faire votre propre projet par vous-même.

Échéance

Ce projet est à remettre à l’échéance prescrite dans le pigeonier correspondant sur eCité.

Tout projet remis en retard ne sera pas corrigé.

Remise

Vous devez remettre ce projet dans un fichier .zip, .rar ou .7z contenant l’ensemble de votre projet web solutionnant cet énoncé.

Si vous ne respectez pas cette consigne, le projet ne sera pas corrigé. Assurez-vous aussi de remettre le bon fichier.

Description

Description du projet

Ce projet consiste à créer un petit jeu de style arcade qui pourra se jouer dans votre navigateur web. Le jeu permet de déplacer un véhicule (un tank) dans un labyrinthe de blocs et de trésor. Quand le véhicule trouve un trésor, il gagne des points. Si le véhicule se déplace et brise des blocs, il perd de l'essence. Le but est simplement d'amasser le plus de points possibles avant que le véhicule n'ait plus d'essence.

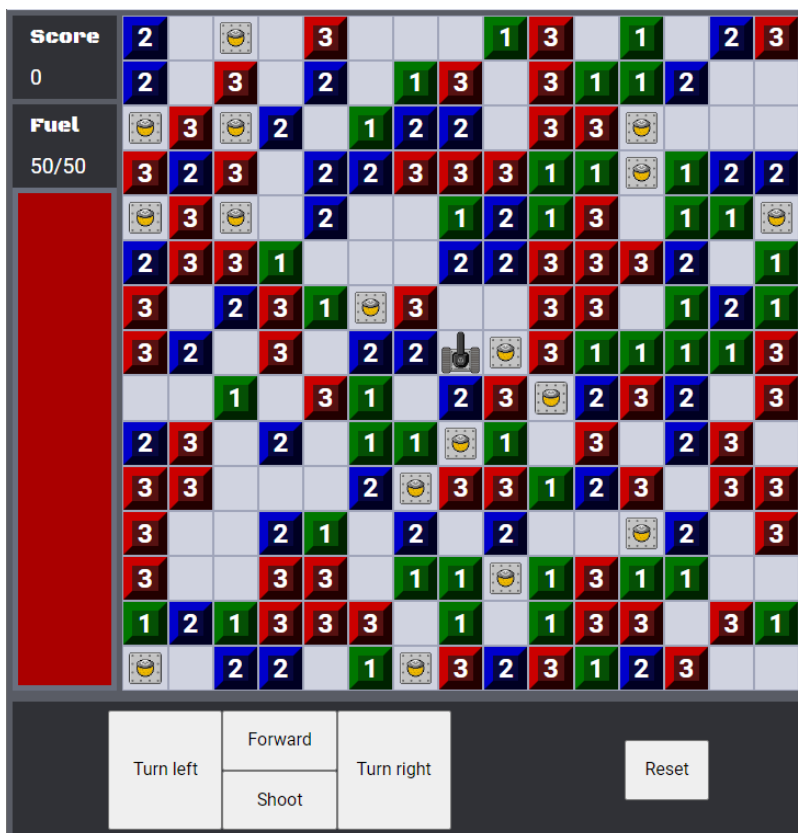


Image à titre indicative seulement. Si vos projets sont un peu différents, c'est correct!

Génération du labyrinthe

Le labyrinthe est représenté par une grille à 2 dimensions. Le contenu de la grille sera généré aléatoirement, mais sa taille est fixe. Sa largeur et hauteur doivent être de 15 tuiles.

Par défaut, chaque tuile de la grille du labyrinthe peut contenir une des 4 valeurs suivantes à probabilité égale :

1. Une tuile vide
2. Une tuile avec un bloc fragile (se brise en un seul tir)
3. Une tuile avec un bloc moyen (se brise en deux tirs)
4. Une tuile avec un bloc solide (se brise en trois tirs)

Véhicule du jeu

Le véhicule de votre jeu doit commencer sur la tuile du milieu de la grille du labyrinthe. Si cette tuile contient un bloc enlevez-le pour y mettre une tuile vide.

Le véhicule possède une direction et peut seulement se déplacer vers l'avant d'une seule tuile à la fois quand on clique sur le bouton pour avancer. Pour changer la direction du véhicule, vous offrirez au joueur 2 boutons pour tourner le véhicule vers la gauche ou la droite. En changeant la direction du véhicule, vous pourrez le déplacer vers l'avant dans une direction différente. Si le véhicule essaye de se déplacer vers une tuile contenant un bloc, le déplacement est annulé. De la même façon, si le véhicule essaye de se déplacer à l'extérieur du labyrinthe, vous devez annuler le déplacement.

Le véhicule pourra tirer les blocs devant lui. Vous devez offrir un bouton permettant au véhicule de tirer. En tirant sur un bloc, celui-ci se brisera un peu. Les blocs solides qui sont tirés deviennent des blocs moyens. Les blocs moyens tirés deviennent des blocs fragiles. Les blocs fragiles tirés deviennent des tuiles vides sur laquelle le véhicule peut se déplacer.

Chaque déplacement et tir du véhicule dépensera de l'essence. Le véhicule commencera avec une quantité d'essence de 50 et cette quantité diminue de 1 à chaque tir ou déplacement. L'action de tourner le véhicule ne diminuera pas la quantité d'essence.

Trésors

Le labyrinthe doit contenir 20 trésors. Ceux-ci sont générés à des emplacements aléatoires dans la grille du labyrinthe et être affiché à l'utilisateur.

Deux trésors ne peuvent pas être générés au même endroit. Si cela arrive, vous devez régénérer un nouvel emplacement pour le trésor en question. De la même façon, un trésor ne peut pas être généré sur la tuile de départ du véhicule.

Si le véhicule avance sur une tuile de trésor, cette tuile deviendra une tuile vide et le joueur gagnera des points

Pointage

Le jeu doit conserver un score pour le joueur et lui afficher en tout temps. Dépendant des actions du joueur, il gagnera ou perdra des points :

- Si le véhicule tombe sur une case de trésor, il gagne 1000 points
- Pour chaque case vide sur laquelle le véhicule avance, il perd 15 points
- Pour chaque tir du véhicule qui abîme ou détruit un bloc, il perd 10 points

Affichage

L'affichage du labyrinthe doit être clair à l'écran. Vous devez voir la position des blocs et de leur durabilité, des trésors ainsi que du véhicule.

L'essence restante doit être affichée au joueur. Il n'est pas nécessaire de l'afficher sous le format d'une barre d'énergie, mais vous devez au moins l'afficher de façon textuelle (ex : « 38/50 »).

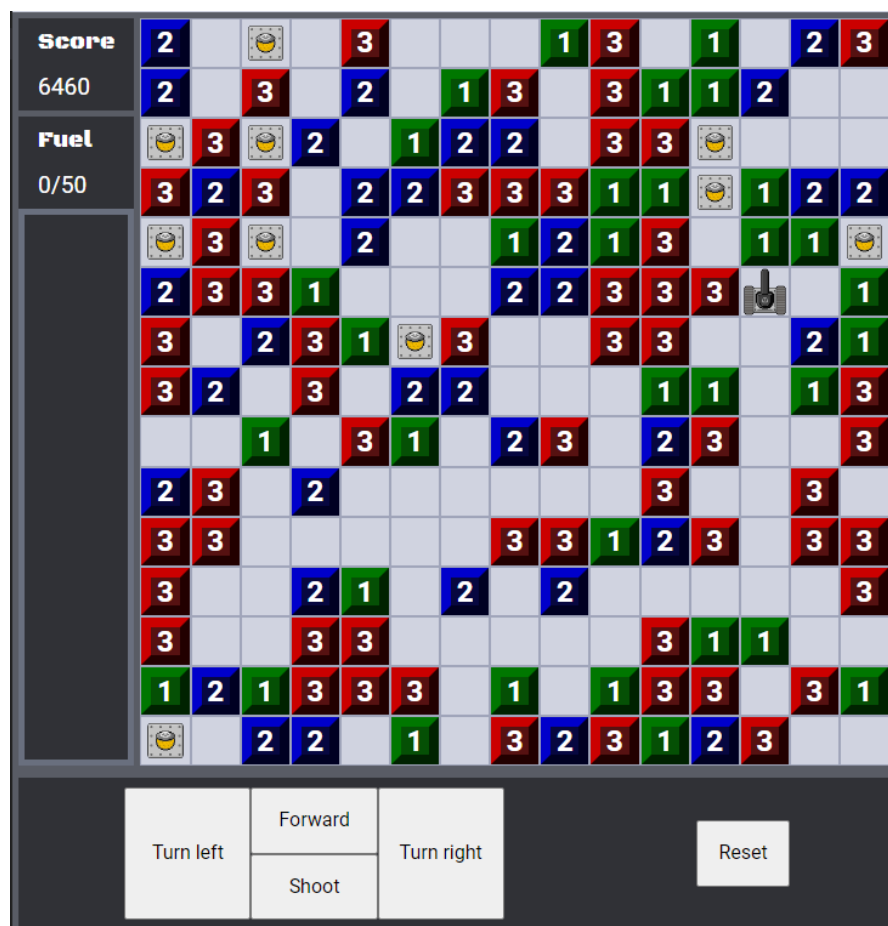
Le score doit être affiché sous un format numérique et doit être clairement visible à chaque étape de votre programme.

Vous pouvez vous servir de l'image ci-dessus comme inspiration pour l'interface graphique, mais vous pouvez aussi la faire de façon différente.

Fin du programme

Le jeu se termine quand le véhicule n'a plus d'énergie. Lorsque c'est le cas, le joueur ne peut plus se déplacer ou tirer.

Assurez-vous d'avoir un bouton fonctionnel pour inviter le joueur à redémarrer le jeu. Pour redémarrer le jeu, **vous ne devez pas rafraîchir la page**, mais simplement réinitialiser les données du jeu.



Fonctionnalités supplémentaires

Pour ce projet, je vous encourage fortement d'essayer d'ajouter des fonctionnalités supplémentaires. Il sera possible d'accumuler des points bonis pour le projet en fonction de la complexité de vos ajouts au projet. Voici une liste d'exemples de fonctionnalités qui pourrait vous donner des points supplémentaires :

- Changer la génération du labyrinthe pour ajouter des fonctionnalités supplémentaires, comme des blocs indestructibles ou des réservoirs d'essence.
- Ajouter un inventaire au véhicule. Le véhicule pourrait trouver des objets dans le labyrinthe qui lui permettrait d'activer certains effets, comme détruire des blocs autour de lui, automatiquement amasser les trésors autour du joueur ou encore de se déplacer de plusieurs cases en passant au travers des blocs. Ces objets ou pouvoirs pourraient aussi baisser la quantité d'essences à leur utilisation
- Ajouter des véhicules ennemis qui se déplace et brise des blocs automatiquement à chaque déplacement ou tir du joueur. Les véhicules ennemis tombant sur une tuile de trésor détruiraient le trésor. De la même façon, si un ennemi tombe sur le joueur, ils diminuent sa quantité d'essence.

Si vous ajoutez ce genre de fonctionnalités, essayez de garder le jeu bien équilibré pour qu'il ne devienne pas trop facile ou simplement impossible. N'hésitez pas à contacter l'instructeur pour savoir si une fonctionnalité est assez intéressante ou non.

Interdiction

Vous pouvez effectuer des recherches sur le web pour trouver comment faire des opérations simples, mais tout code provenant clairement d'un autre exemple en ligne sera considéré comme de la fraude scolaire.

Les « canvas » en HTML et les contextes graphiques en Javascript ne sont pas autorisés pour le projet puisque ceux-ci ne vous font pas utiliser des concepts sur lesquels vous êtes évalués.

Pondération

Critères

Ce projet vaut pour 35% de la session. Toutefois, il sera évalué sur 36 points. Plus précisément, vous serez évalué selon les critères suivants :

Critères	Pondération
Génération du labyrinthe et des trésors	15 points
Déplacement du véhicule	15 points
Affichage et gestion du score et de la quantité d'essence	4 points
Redémarrage du jeu	2 points
Total	36 points
Fonctionnalité supplémentaire	4 points boni maximum

Pénalités

Assurez-vous de bien indenter votre code (formatage), documenter votre code (commentaires), de suivre les bonnes pratiques et aussi de faire un programme facile à utiliser pour l'utilisateur. Vous pouvez perdre jusqu'à 20% des point du projet si vous ne suivez pas ces consignes.

Voici quelques exemples de bonnes pratiques à respecter. La liste n'est toutefois pas exhaustive.

- Ne pas utiliser la fonction alert()
- Ne pas utiliser les vieilles façons de détecter les événements
- Utiliser les modules ES et non le code Javascript sans module
- Séparer le code dans différents fichiers
- Séparer le code en plusieurs fonctions (ne pas avoir de fonction ou de bloc de code trop gros)
- Mettre un commentaire de documentation au-dessus de chaque fonction
- Mettre un commentaire de documentation au-dessus de chaque variable importante
- Les noms de variables doivent suivre la même nomenclature que le Java ou C#
- Les noms de fonction doivent suivre la même nomenclature que le Java
- L'indentation du code doit être parfaite
- Ne pas mettre plusieurs lignes vide de suite dans le code
- Ne pas avoir d'erreurs ou warnings dans la console Javascript du navigateur

Astuces

Il peut être intimidant d'avoir à créer un programme comme celui-ci. Voici quelques trucs et astuces pour vous aider :

- Ne commencez pas par l'interface graphique. Dans ce genre de programme, il est plus facile de s'occuper des données en mémoire et de leur modification. Modifiez l'interface graphique uniquement lorsque les opérations en mémoire fonctionnent bien.
- La génération du labyrinthe est une grille à 2 dimensions. Elle doit donc être un tableau à 2 dimensions en mémoire. Ce tableau contiendra des valeurs indiquant ce qu'il y a pour chaque tuile du labyrinthe, soit une tuile vide, un bloc de différentes solidités ou un trésor. Ces valeurs peuvent être des entiers (0 pour tuile vide, 1 pour bloc fragile, 2 pour bloc moyen, etc...), des caractères ou même du texte. À vous de décider comment vous voulez le programmer.
- Séparez certains concepts dans différents fichiers. Il est plus facile de comprendre le code et de bien s'organiser si vous séparer les différents éléments de votre jeu dans différents fichiers Javascript dès le départ. Dans notre cas, on peut probablement séparer la grille du labyrinthe, le véhicule et la logique principale du jeu dans leur propre fichier. N'hésitez pas à vous créer plus de fichiers si c'est nécessaire.
- Commencez par faire une interface simple. L'affichage dans le navigateur peut simplement être des <div> de différentes couleurs pour identifier le joueur, les pièges et les trésors. Rien ne vous empêche par la suite d'améliorer votre interface.