ROUICHI Adil

Projet Moteur de Jeu

DEBBAH Rabah

Introduction :

Suite à notre présentation orale nous avons décidé de repartir presque de zéro afin de pouvoir mettre en place une génération procédurale de circuit satisfaisante.

Nous avons créer trois classes différentes afin de réaliser notre projet plus aisément :

→ La classe Cars : Correspond à un objet de type voiture.
→ Elle comporte toutes les caractéristiques de la voiture (vitesse, vitesse de freinage, angle de virage, sa position, son angle…) et permet de gérer son déplacement, ses collisions, son comportement et son rendu.

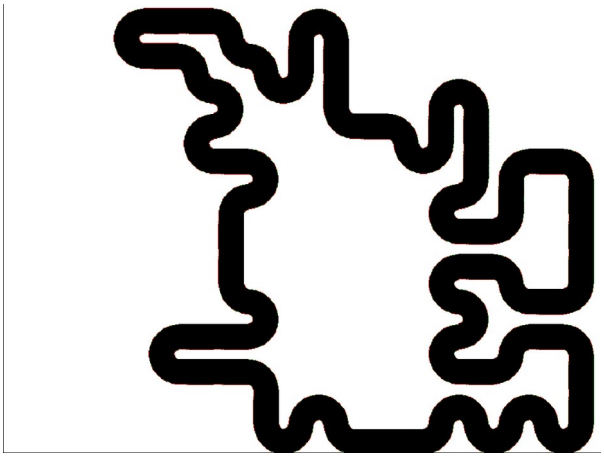
→ La classe Tracks : Correspond à un objet de type circuit.
→ Elle comporte toutes les caractéristiques du circuit (sa taille, son tableau de valeurs, sa case de départ…) et permet de gérer sa création et son rendu.

→ La classe Game : Correspond au jeu et contient les autres objets.
→ Elle comporte le canva, le contexte, le circuit, les voitures, le fond et les décors. Elle permet de générer les décors, le fond et fait le lien entre les autres objets.

Notre projet utilise 40 sprites différents afin de créer un circuit, son terrain et ses décors de manière procédurale pour permettre à des voitures aux caractéristiques différentes de faire la course en toute autonomie.

Musique : Le thème du jeu a été réalisé par nos soins via le logiciel FL Studio afin de pouvoir accompagner ce jeu.

Génération procédurale :

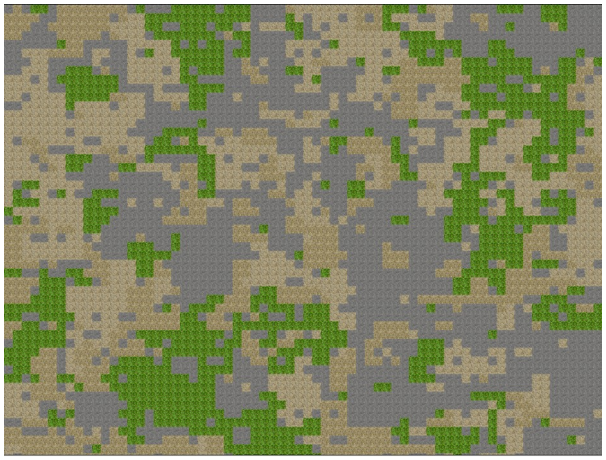


La première étape de notre génération de carte procédurale est la génération du circuit. Pour ce faire nous créons un tableau de taille (w*h) initialisé à 0.
Nous choisissons x et y les coordonnées d'une case de départ choisie aléatoirement.
Puis nous appelons next_road(x,y,1) :
function next_road(x,y,acc)

-tab[x,y] = acc;
-quelles cases permettent de revenir à la case de départ sans passer par une case > 0 ?
-Parmi ces cases, choisir aléatoirement une case adjacente à tab[x,y] appelée ici x' et y'.
-Aucune case adjacent disponible ? Stop
-Sinon : next_road(x', y', acc+1)

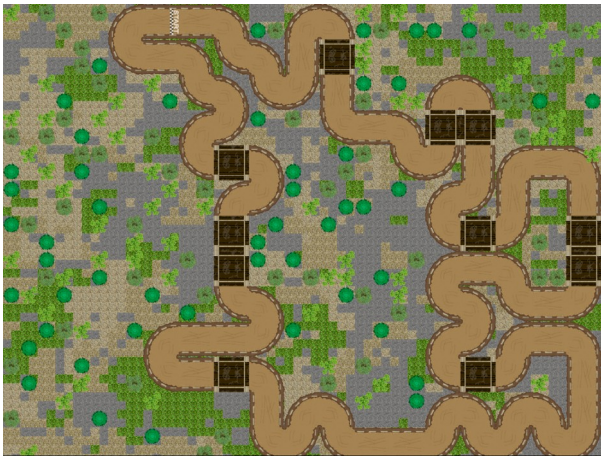
Une fois cela fait, nous avons un tableau avec des valeurs inférieures à zéro (cases vides) et des valeurs supérieures à zéro (routes). Les cases qui correspondent à des routes ont des valeurs s'incrémentant de 1 en 1 afin de pouvoir ensuite reconstituer la route. Pour ce faire, nous parcourons le tableau et à chaque case ayant une valeur n supérieure à 0, nous cherchons où sont situées les cases qui ont pour valeur (n-1) et (n+1) par rapport à celle-ci. Grâce à cela, nous pouvons attribuer à cette case une valeur correspondant au type de route (0 : ligne droite verticale, 3 : corner haut droit…).

On a maintenant notre route. Nous la dessinons tout d'abord en noir et blanc lors de l'initialisation du jeu afin de pouvoir stocker les pixels dans un tableau qui nous permettra de savoir où se trouve la route afin de gérer le comportement de nos voitures.



Ensuite, nous créons le fond sur un tableau de taille (w*h*4) afin d'avoir plus de détails. Pour créer le background nous effectuons ces étapes :

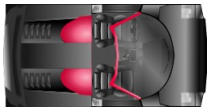
(Tant qu'il y a des cases vides) :
-Choisir une case vide aléatoirement.
-Changer la valeur de remplissage (herbe, sable…).
-Choisir un nombre de cases à remplir aléatoire entre 0 et le nombre de cases vides restantes divisé par 2.
-Remplir à partir de cette case une case vide adjacente puis recommencer depuis cette nouvelle case tant qu'il y a des cases vides adjacentes et que le nombre de cases à remplir n'est pas atteint. Ensuite recommencer la boucle.



Enfin, nous créons le décor sur un tableau de taille (w*h*2). Pour ce faire nous déterminons un nombre de décors à disposer sur notre tableau par rapport aux cases ne comportant pas de route. Nous remplissons le tableau avec des valeurs de décors aléatoires tant que nous n'avons pas atteint le nombre de décors à disposer.

Afin d'habiller les routes nous avons décider d'y mettre des toits pour créer des tunnels. Nous établissons un nombre de tunnels à générer par rapport aux nombres de routes en ligne droite puis nous les disposons dessus de manière aléatoire.

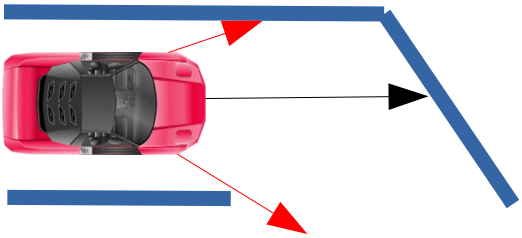

Les voitures :




Nos voitures disposent de diverses caractéristiques :

→ Accélération, vitesse maximum, vitesse de freinage, angle de virage…

Les valeurs de ces caractéristiques diffèrent selon le type de voiture et permet tout d'abord d'avoir une conduite qui diffère d'une voiture à l'autre mais également d'avoir un comportement différent de la part du pilote.



En effet, la voiture dispose d'une « vision » qui est composé de trois vecteurs qu'elle projette, un vers l'avant et deux en diagonale.
La voiture décidera de tourner selon la différence entre ses deux vecteurs de vision diagonaux (elle choisira le plus grand) et sa capacité à tourner.

De même pour le freinage ou l'accélération qui dépendent de la distance entre l'avant de la voiture et un obstacle, de sa capacité à freiner et de sa vitesse.

Ces différents aspects permettent d'apprécier une course qui peut être assez divertissante et aux résultats différents selon les circuits (beaucoup de virages, beaucoup de lignes droites…).

Collisions :

Nous avons mis en place un système de collision assez simple qui détecte une collision si l'un des 4 coins de la voiture se trouve hors de la route. Cependant, du fait de la conduite exemplaire de nos voitures, les collisions n'arrivent jamais. De plus, nous avons, sur la fin de la réalisation de notre projet, eu un problème quant aux collisions qui fait que les voitures ne redémarrent pas.

Conclusion :

Pour conclure, nous tenons à dire que nous sommes plutôt contents du rendu de notre jeu. Nous trouvons la génération procédurale variée et efficace. Nous aurions aimé pouvoir ajouter des sons de voiture et un peu d'aléatoire dans le comportement des voitures afin de faire varier encore plus les courses. Nous espérons que notre projet vous plaira malgré notre retard et que vous vous prendrez comme nous, au jeu de savoir quelle voiture arrivera la première.

Nous vous remercions pour le temps accordé à la lecture de ce rapport de projet.

PS : Nous avons une bogue sur notre programme qui arrive parfois lors du premier lancement du jeu. Les voitures ne démarrent pas, il suffit de rafraîchir la page.