

Garden Simulator

Pour le projet de C++, vous allez devoir développer un jeu de gestion dans lequel on simule un jardin, dans lequel travail un groupe de jardiniers, avec pour objectif de faire pousser des fleurs. Ce qui suit est une base de règles, que vous pouvez enrichir si vous le souhaitez.



Exemple d'affichage

Les plantes

Il y a plusieurs type de plantes : des fleurs, des grains, des legumes

Une plante a une durée de pousse, exprimé en secondes. C'est le temps nécessaire pour passer du stade de jeune pousse jusqu'au stade de plante complètement développée.

Une plante a une date de plantation, exprimée en secondes, qui permet de retenir à quel moment elle a été plantée.

Ces deux informations permettent de déterminer l'étape de pousse, comprise entre 0 (jeune pousse) et 4 (plante mature). Ce numéro d'étape servira à faire évoluer l'affichage des plantes.

Certaines plantes peuvent être récoltées (plantes à grain et légumes), d'autres plantes ne sont pas destinées à la récolte.

Idéalement, chaque type plante devrait avoir au moins 2 variétés avec des caractéristiques différentes : les radis peuvent pousser plus vite que les fraises, le café peut donner plus de grains que le blé, etc.

Une plante devrait posséder toutes les images nécessaires à son affichage.



Les 5 étapes de croissances d'un plant de tomate

Les fleurs

Faire pousser des fleurs est le but ultime de tout jardin. Elles n'ont pas d'utilité. Elles ne peuvent pas être récoltées. Une fois à maturité, elles restent indéfiniment en jeu.

Les plantes à grains

Faire pousser des plantes à grain, et les récolter, vous permet de faire du commerce avec le vaste monde. Quand une plante à grain est arrivée à maturité, un jardinier peut récolter des grains. Les grains sont la monnaie du jeu qui permet d'acheter des semences (de fleur, de plantes à grains et de légumes) et de recruter de nouveaux jardinier. Ce sera à vous de fixer les prix pour régler la jouabilité de votre jeu.

Les légumes

Faire pousser des légumes, et les récolter, vous permet de nourrir vos jardiniers, ce qui modifie leur comportement.

Quand un légume est arrivée à maturité, un jardinier peut le manger.

Les jardiniers

Les jardiniers sont vos ouvriers, ce sont eux qui récoltent les grains. Ils agissent automatiquement. Chaque ouvrier est repéré par sa position.

Chaque jardinier a un niveau de moral, et se souvient de l'heure à laquelle son humeur a changé. Juste après avoir mangé, il est « content ». Il est plein d'entrain pour travailler : son seul objectif est de trouver la plante à graine à maturité la plus proche pour aller la récolter.

A bout d'un certain temps, son moral diminue à un niveau « normal ». Il hésite entre travailler et manger : il se dirige vers la plante à maturité la plus proche, que ce soit une plante à grains ou un légume (les fleurs ne l'intéressent pas).

A bout d'un certain temps sans manger, son moral empire encore, il devient « mécontent ». Il se désintéresse totalement de la récolte de grains : il se dirige vers le légume à maturité le plus proche pour le manger. Si il n'y a pas de légume à maturité en jeu, il cherche la fleur la plus proche pour la détruire (révolution!!!). Si il détruit une fleur, son moral remonte au niveau « normal ».

Un ouvrier content ou normal ramasse les grains si il est au même endroit qu'une plante à grains (à maturité).

Un ouvrier normal ou mécontent mange si il est au même endroit qu'un légume (à maturité).

Un ouvrier mécontent détruit une fleur il est au même endroit qu'une fleur (à maturité).

Le niveau de moral du jardinier doit être visible à l'écran, afin de permettre au joueur de savoir si ses ouvriers sont correctement nourris. Vert indique un jardinier content, jaune un jardinier normal, rouge un jardinier mécontent.

Pour cela, chaque jardinier devrait posséder toutes les images nécessaires à son affichage.



Planche d'animation d'un jardinier content (vert)

Le champs

Le champs est une grille dont chaque case peut contenir :

- rien, c'est juste de la terre ou de l'herbe sans intérêt.
- une plante, de n'importe quel type

Les jardiniers se déplacent librement dans les champs. Plusieurs jardiniers peuvent avoir les mêmes coordonnées.

Le jardin

Le jardin est l'entité qui englobe toutes les informations sur le jeu, il contient l'ensemble des jardiniers, le champs, votre stock de grains, etc. Toute information sur l'état du jeu doit être accessible depuis le jardin.

Le joueur

Le joueur peut interagir avec le jardin à l'aide de la souris et/ou de raccourcis clavier.

Il peut ajouter des jardiniers (si il a assez de graines) ou planter des fleurs, des legumes ou des plantes à graine à l'endroit de son choix, à condition que le champs soit vide à l'endroit voulu.

En particulier, il ne peut pas ajouter un jardinier si il y a un jardinier aux coordonnées de la case où se trouve la souris.

Graphismes

Des fichiers d'images sont fournis, ils sont issus de :

<https://opengameart.org/content/green-cap-character-16x18>

<https://opengameart.org/content/farming-crops-16x16>

Le licences de ces fichiers sont disponibles à ces adresses.

Vous pouvez les utiliser, ou en utiliser d'autres.

Vous pouvez n'en utiliser qu'une partie : vous n'êtes pas obligés de gérer 20 types de plantes.

Vous pouvez les modifier, par exemple avec l'outil Gimp.

L'animation des Jardiniers peut être chronophage pendant le développement. Si vous voulez aller vite, représentez chaque jardinier avec une image fixe, et déplacez-le sur la grille de case en case. Si vous avez le temps, animez les jardiniers et permettez-leur de se déplacer de manière fluide dans le jardin.

Gameplay

Vous avez toute latitude pour fixer les divers éléments du jeu : temps nécessaire pour que chaque type de plante pousse, taille du champs, taille de l'affichage, le nombre de jardiniers au démarrage d'une partie, le coût en graine de jardiniers supplémentaires et des différentes plantes, etc. Ceci vous permettra de gérer la difficulté du jeu, de faire des tests en changeant ces valeurs, etc.

Cependant, ces valeurs doivent IMPERATIVEMENT être définies de manières centralisées dans un fichier (fichier texte ou .h). Il ne doit y avoir aucun « magic number » dans votre code source.

Organisation du projet

Modélisation UML

Vous devez commencer par modéliser votre projet en UML.

Cette étape est importante, vous devez anticiper tous les attributs des classes et les types de données manipulées, définir les méthodes les plus importantes pour les classes. Avant de commencer à coder, vous devez pouvoir répondre à des questions telles que :

- comment est représenté le champ ?
- y a-t-il une plante à maturité dans une case donnée du champs ?
- comment connaître le niveau de moral d'un jardinier ?
- etc.

Documentation

Votre code source doit être commenté afin de générer automatiquement une documentation, par exemple avec doxygen.

Commentez au fur et à mesure que vous développez.

Fichiers du projet

Le projet doit être divisé en bibliothèques (.h, .cpp et makefile).

Vous devez utiliser des conventions de nommages standards.

Encapsulation et forme canonique

Les classes doivent être encapsulées.

Les méthodes de la forme canonique ne sont obligatoires que si leur écriture est nécessaire au bon fonctionnement de la classe.

L'écriture de constructeur paramétrés n'est pas obligatoire, sauf si cela vous est utile.

Notation

La notation dépendra de ce que vous aurez réalisé à l'issue du projet, ainsi que de propreté du code, et en particulier des commentaires et de la documentation.

Un barème différent sera appliqué selon la taille du groupe (1, 2 ou 3 personnes).

Par exemple, si vous êtes seul, les jardiniers peuvent se déplacer de case en case, comme des pièces sur un échiquier.

Si vous êtes deux, le déplacement des jardiniers devrait être fluide quand ils se déplacent.

Si vous êtes trois, non seulement le déplacement devrait être fluide, mais le personnage devrait être animé en fonction du temps et de la direction de son mouvement.

Rendu

Le projet est à rendre par mail, au plus tard le dimanche qui suit la dernière séance du cours de POO. La remise se fera sous la forme d'un fichier zippé contenant l'intégralité de vos fichiers. Si vous avez travaillé en groupe, le mail devra rappeler les noms de tous les membres du groupe.