

Zhao Yuge

Mailly Charles

### Sujet n° 5 : Labyrinthe

## Spécifications

### Objectifs :

- Souris qui trouve le fromage par le chemin le plus rapide
- Plusieurs souris
- Affichage graphique avec SDL2
- Chat
- Création de labyrinthe ?

### Fonctionnalités détaillées

- 1) Chaque souris aura une intelligence qui lui permettra de trouver le fromage de la façon la plus rapide, tout en évitant le chat s'il y en a un.
- 2) Les labyrinthes pourront être chargés depuis un fichier ou générés par un algorithme.

### IHM

- Choix du nombre de souris
- Choix du nombre de chat
- Choix pour le labyrinthe (Génération/Chargement depuis un fichier) avec en option la taille du labyrinthe
- Le fait de cliquer sur une case vide dans le labyrinthe y ajoutera un fromage, s'il n'y en a pas déjà un.

## Conception détaillée

### Structures

- Coords : Gère une position dans le labyrinthe
- Chunk : Labyrinthe ou portion de labyrinthe
- Mouse : Gère une souris
- Cat : Gère un chat
- Cheese : Gère un fromage
- Node : Utile pour trouver le meilleur chemin
- Array : Contient un tableau ainsi que sa taille utilisée

Nous sommes susceptibles de rajouter des structures en fonction des besoins de la SDL2.

### Fichiers de données

Le seul type de fichier que nous allons utiliser sauvegardera des labyrinthes.

### Fichiers et fonctions

- Utils.h/c, --> Nous deux en fonction de nos besoins  
Permettant d'être inclut partout afin d'utiliser la structure Coords ainsi que les structures Array qui seront employées par la majorité des autres structures
- Chunk.h/c --> Charles  
Fichiers regroupant la structure pour un labyrinthe et toutes les fonctions utiles
  - o Chargement d'un labyrinthe avec nom de fichier en paramètre
  - o Génération d'un labyrinthe avec taille en paramètre
  - o Sauvegarde d'un labyrinthe
  - o Affichage d'un labyrinthe
  - o Fonction permettant de dire si une case (x,y) est un mur ou pas
- Entities.h/c --> Yuge  
Fichiers regroupant les structures Mouse, Cat et Cheese ainsi que des fonctions pour les manipuler facilement
  - o Affichage des différentes structures
  - o Création des différentes structures
  - o Gestion du déplacement des différentes structures
- Path.h/c --> Yuge  
Fichiers contenant la structure Node, les fonctions pour manipuler les Nodes et contenant l'algorithme pour retrouver le chemin vers le fromage
  - o Fonction convertissant une Node en un tableau de Coords
  - o Fonction trouvant le meilleur chemin en fonction de l'arrivée et du départ ainsi que du labyrinthe utilisé (pour les murs)
- Application.h/c --> Charles  
Fichiers gérant l'affichage et la boucle principale (Evènements utilisateur/Mise à jour/Rendu)
  - o Fonction gérant la boucle principale
  - o Fonction traitant les entrées de l'utilisateur
  - o Fonction mettant à jour les entités (Souris/Chat/Fromage) pour les déplacements et l'affichage
  - o Fonction effectuant l'affichage

## Planning

- Séance 2 : Installation de la SDL2, création des fichiers, des structures
- Séance 3 : Entités pour Yuge et Application pour Charles
- Séance 4 : Path pour Yuge et Chunk pour Charles
- Séance 5 : Connecter tout et vérifier

## Github

Utilisation de Git et GitHub à cette adresse : <https://github.com/Cmdu76/ProjetEsigelec>