

Projet C avancé

Labyrinthe

BOCQUET Joanes

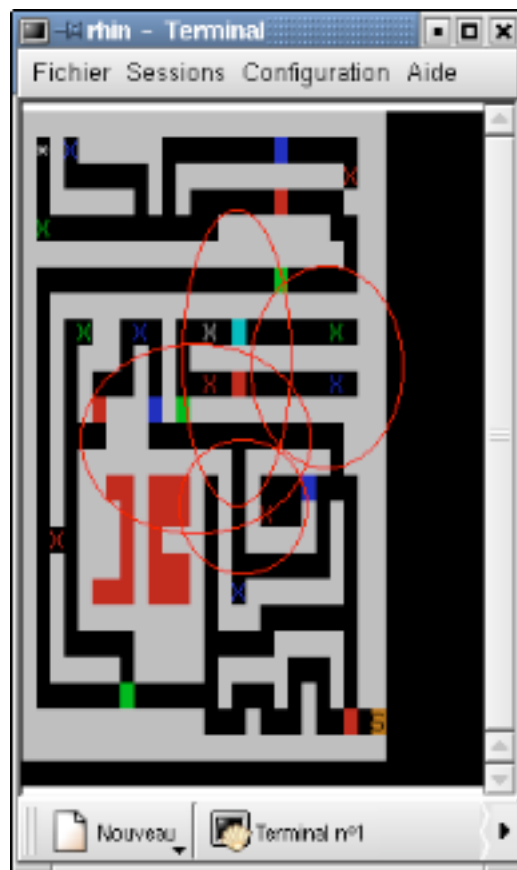
INTRODUCTION

L'objectif du projet est de programmer un petit jeu de labyrinthe. Le personnage doit arriver à la sortie du labyrinthe, mais ce labyrinthe comporte différentes portes de couleur qu'il faudra ouvrir. Les clés des portes sont réparties dans le labyrinthe. Une clé est perdue dès qu'elle a ouvert une porte et la porte reste ouverte jusqu'à la fin du jeu.

Le deuxième objectif du projet est d'utiliser des bibliothèques existantes. La gestion "graphique" se fait grâce à la bibliothèque ncurses et la gestion du personnage grâce à la bibliothèque du premier projet.

Le jeu et ses règles

Le labyrinthe est en fait une matrice avec en son intérieur des chiffres représentant les murs les clés les portes et la sortie. ce qui donne :



Les regles sont simples, le but est d'arriver a la sortie tout en passant par les portes ce qui implique l'obtention de certaines clés.

Les touches utiles:

- les fleches pour se déplacer
- la touche x pour quitter
- la touche p pour prendre la cle
- la touche l pour visualiser le trousseau de clés

Structure du projet

Fonctions

```
void curses_init();
qui initialise les couleurs de differentes methodes de curses

void dessine_cles(int,int,int);
qui dessine les cles ou la sortie avec les couleurs specifiees a
l'endroit specifie
void draw_Lab();
qui dessine le labyrinthe en separant les murs des cles

char *fixer_Couleur(int,char*);
Renvoie le nom de la cle en fonction de la valeur lue dans matrice du
labyrinthe
int show_Screen(void*,void*);
Destinee a la fonction ListIter qui demande le "format"
int gere_cle();
action de prendre la cle, ajoute cette cle au trousseau de cles
void show_prot_cle();
Affichage du porte cles
int cleat_porte_cle();
Permet d'effacer l'affichage du porte cles
void deplacement(int,int);
Gere le deplacement du curseur
Gere le redessinage des cle si non prises
void dplcmt_possible(int,int);
ProcEDURE qui gere le deplacement a venir
* teste si le deplacement est possible
void clavier(int);
qui gere les touches utilisées pour le déplacement
```

Structure de données

Le labyrinthe est donc représenté comme un tableau a deux entrées. Les clés, murs, portes et la sorties sont représentés comme des entiers differents, et sont ammenés a changer lorsque la clé a ete prise ou bien que la porte a ete ouverte.

Pour le porte cles on utilise une liste circulaire d'elements afin de pouvoir accumuler les clés obtenues.

L'affichage

L'affichage du tableau s'effectue en debut de programme et les cases qui changent seules sont modifiées ce qui optimise assez l'affichage.

Ajouts et idées

Le projet comporte quelques modifications légères par rapport au sujet initial. Mes clés sont de la couleur de la porte qu'elles ouvrent, ce qui rend plus facil le parcours du labyrinthe. Ce qui peut etre un premier niveau assez simple.

Le but de mon projet etait de faire un labyrinthe avec plusieurs niveaux et pouvoir y ajouter des montres. Aussi il etait possible de faire accelerer le curseur afin de se déplacer plus vite. Ou bien de faire un générateur de niveaux aléatoires.

Mais le temps ne m'a pas permis de concretiser mes objectifs....

Donc le travail impératif a été effectué:

- Jouabilité normale
- Compilation correcte
- Ce dossier explicatif
- Makefile