

Projet Slitherlink

Matias VECCHIO ,Kevin XU

Introduction

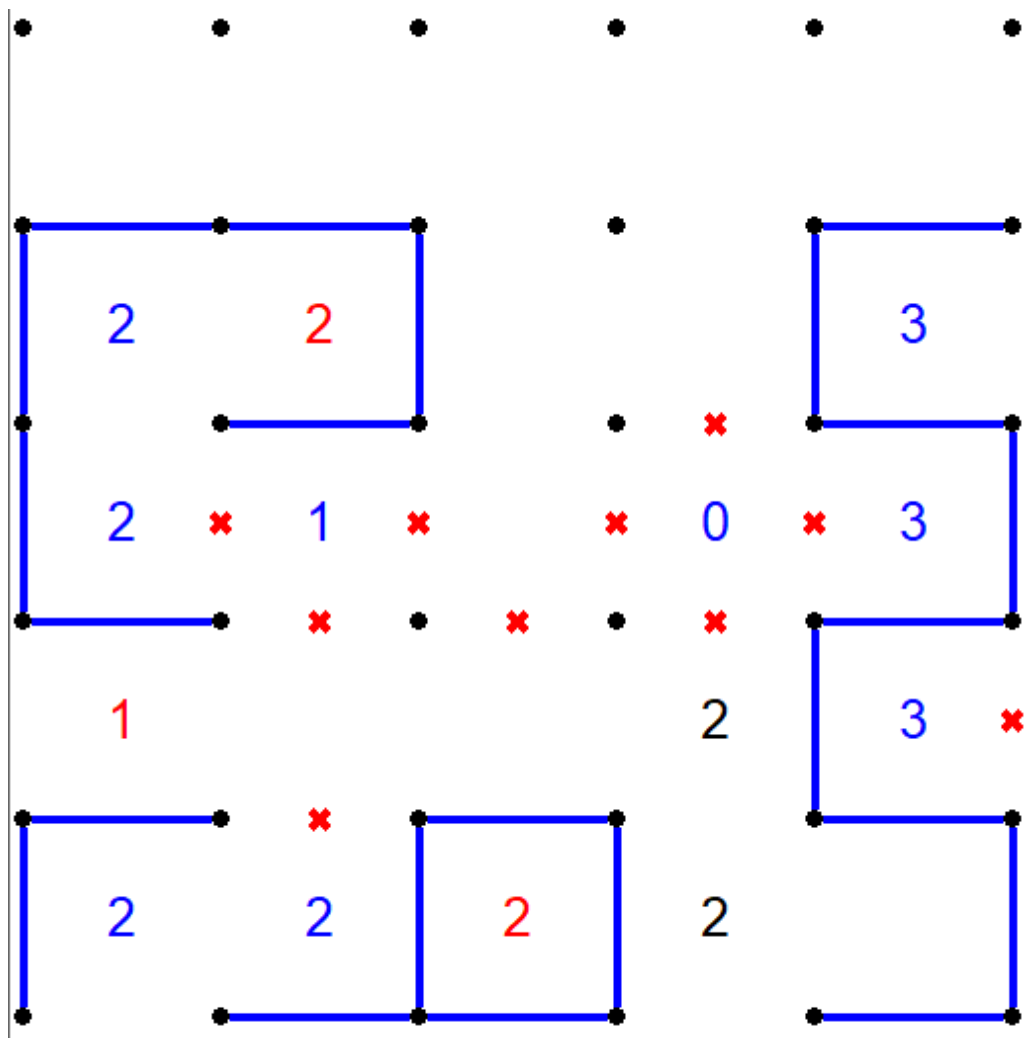
Slitherlink est un jeu ,où on a une grille de X par X le but est remplir les conditions de chaque case avec un nombre ,tout en les reliant pour que l'ensemble soit fermé.

Guide Utilisateurs

Il faut tout d'abord choisir la grille que l'on veut en cliquant sur le rectangle du triangle voulu avec autres qui permet de lire des fichiers supplémentaire .



Il suffit de cliquer sur les segment horizontal a coté des cercles , chaque segment est lier a de 2 case ,les cases vide non pas de condition de nombre de segment , les cases avec des nombre est le nombre de segment a avoir.



Pour activer le solveur il vaut appuyer sur SHIFT pour le solveur graphique et ENTRER pour le non graphique

État d'avancement et répartition du travail

Tache 1 : Structures de donnée

fait par Kevin XU puis modifié Matias VECCHIO pour simplifier la tache 2

Terminer

Tache 2 :conditions de victoire

fait par Matias VECCHIO

Terminer

Tache 3 :Interface graphique

Fait par Kevin XU

Terminer

Tache 4 :Recherche de solutions

fait par Matias VECCHIO

Terminer

Démarche programmation

Nous avons fait la tache 3 en même temps que la tache 1 car elle simplifier garce a la visualisation des erreur, puis modifié la tache 1 pour la tache 2 pour simplifier la tache 2.

La tache 3 fut embêtante a faire car il fallait trouver comment prendre les seulement les côtés , nous avons réussi a le faire en faisant un schéma et regardant le pourquoi et comment de chaque trait vertical et horizontal.

Pour la tache 4,nous avons fait le solveur naïf puis l'avons améliorer en mettant fonction créant une liste de point de depart , et choix par priorité et change de points si le points n'a pas de solution.

Difficulté rencontrer

La grille 1 et 2 est trop long pour le solveur .

Fonction

Tache 1 : Structures de donnée

tableau_init : fonction prennent en paramétré nom(String) pour récupéré ce qui a l'intérieur dans txt et le transformer en liste de liste nommer plateau.

est_trace,est_efface,est_interdit : regarde si le segment est dans l'état demandé renvoie true s'il est dans l'état demander et sinon renvoie false .

tracer_segment,interdire_segment,effacer_segment : Il donne la liste dans segment dans l'état demandé.

statut_case : Lis l'état de la case ,le nombre de segment ou case vide.

Choix_grille : Fonction servant de sélecteur de grille.

Tache 2 : conditions de victoire

longueur_boucle : Donne la longueur de la boucle.

win : la fonction qui gère les conditions de victoire , regarde si chaque case numéroté satisfait les condition (c'est-a-dire que l'indice recupere de la fonction statut_case est zero)et regarde si elle est fermé.

Tache 3 : Interface graphique

Init_graph : fonction graphique récupéré plateau ,en utilisant fltk il crée une fenêtre puis utilise plateau pour faire la grille visuel.

clique : récupère le clique et détermine si c'est un clique droit ou gauche ,puis selon le clique trace un segment ou interdit si la fonction tracer_segment ou interdire_segment renvoie true ,sinon l'efface.

Dessin depart : dessine ou efface le segment (ou la croix).

Tache 4 : Solveur

depart: renvoie un liste de point de départ possible.

poseinterdit : pose les segment interdit pour le solveur.

reset_interdit : efface les segment interdit pour le solveur.

solveur : La fonction solveur (fonction récursive) est un solveur naif c'est a dire essaye tout les possibilité possible.
