RAPPORT EXPLICATIF



du projet d'Info 4A - Labyrinthe en C et C++

PARCOURS MAX

Partie 4B - implémentation de distMin

Contribution personnelle - Génération : Algorithme avec backtracking

Par: Evan PETIT - IE4 I912

Professeur de TP: M A. OUAZZANI

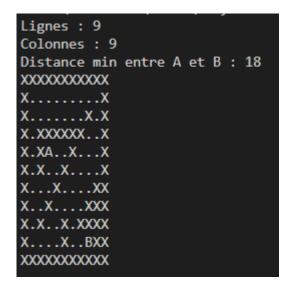
SOMMAIRE

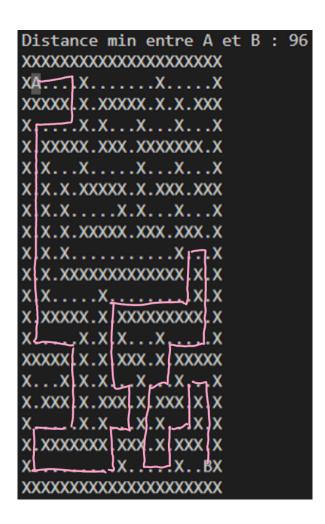
1. Partie 4B

2. Contribution personnelle

Tests effectués

En effectuant quelques tests, on peut vite vérifier que cet algorithme fonctionne correctement. Il suffit de créer quelques labyrinthes à l'aide de descripteurs, ou bien d'en générer avec les algorithmes de génération. Voici deux exemples :





Tester l'algorithme avec backtracking

Il est situé dans la fonction genLabyBack() de la classe labyrinthe.

Pour le tester, il suffit de rentrer dans la fonction main() : labyrinthe* lab = new Labyrinthe(lignes, colonnes); lab -> genLabyBack(); lab -> affiche();

Ainsi on peut le tester. Les labyrinthes peuvent atteindre des tailles phénoménales car l'algorithme en un temps quasi linéaire (on casse un mur par case) + quelques autres constantes (remonter la pile pour le backtracking). Des labyrinthes 1000*1000 sont générés en une poignée de | PS D:\Documents\Cours\Labyrinthe\Partie4> ./Labyrinthe.exe

secondes.

```
PS D:\Documents\Cours\Labyrinthe\Partie4> ./Labyrinthe.exe
X.....X.X.....X
X.....X.X.X.....X...X...X...X.X
X.X.XXX.X.XXXX.X.XXXX.X.XXXX.X.XXXX.X.X
X.X.X.X....X.X...X.X...X.X...X
X.X.X...X....X.X.X.X.X.X.X.X.X
X...X.X.X...X.....X.X.X.....X.X.X.X....X
X.....X.X...X.X...X.X...X.X....X.X....X
. . . . . . . X . . . . . . . . . X . . . . . . . . . . . . . . X
```

```
x.x...x.x.x.x.x...x....x...x...x...x.x...x...x...x...x...x...x...x...x...x...x...x
```