

Depth-First Search (DFS) Maze Generator in C++

Elias Cherkaoui

Översikt

Projektets syfte är att skapa en labyrintgenerator med hjälp av Depth-First Search (DFS) algoritmen. Labyrinten renderas grafiskt med GLFW.

Implementeringsdetaljer

- **Använda Bibliotek:** Projektet använder GLFW-biblioteket för grafisk rendering omfattande. Det inkluderar också standardbibliotek som `<vector>`, `<iostream>`, `<chrono>`, `<random>`, `<algorithm>`, `<math.h>`, `<stack>`, och `<cstdlib>` för olika funktionaliteter.
- **Cellstruktur:** Labyrinten består av celler, representerade av structen `Cell`. Denna struct innehåller egenskaper som `visited` (boolean), `walls` (en array av booleans som representerar väggarnas existens på fyra sidor), koordinater (`x`, `y`) och slumpmässiga färgkomponenter för visuell distinktion (Används ej just nu).
- **Ritfunktion:** Funktionen `drawCell` ansvarar för att rendera varje cell på skärmen. Den använder OpenGL-kommandon för att rita linjer som representerar cellernas väggar och fyller varje cell med den valda färgen.
- **Genereringslogik för Labyrinten:**
 - Programmet initierar ett rutnät av celler, var och en med väggar på alla fyra sidor.
 - Det väljer slumpmässigt en startcell och markerar den som besökt.
 - Med DFS-algoritmen utforskar det iterativt obesökta intilliggande celler. När det flyttar till en granne, tas väggen bort mellan den aktuella cellen och den valda grannen, och sedan markeras grannen som besökt.
 - Algoritmen fortsätter tills alla celler är besökta, och backtrackar vid behov.