

Specyfikacja Implementacyjna programu Maze Solver

Tomasz Rogalski, Jakub Sokolik

25 kwietnia 2023

Streszczenie

Dokument zawiera informacje na temat implementacji programu w C służącego do odnajdywania ścieżki prowadzącej przez labirynt. Znajdują się w nim szczegóły dotyczące narzędzi i środowisk wykorzystanych do tworzenia programu.

Spis treści

1	Cel projektu - streszczenie	4
2	Ustalenia wstępne	4
3	Podział plików	4
3.1	Pliki główne	4
3.2	Pliki testowe	5
4	Implementacja odnajdywania ścieżki	5

1 Cel projektu - streszczenie

Program pozwala na odnalezienie ścieżki, niekoniecznie najkrótszej, od początku do końca labiryntu dostarczanego w postaci pliku tekstowego. Plik ten składa się z następujących znaków:

- P – punkt wejścia do labiryntu
- K – punkt wyjścia z labiryntu
- X – ściana
- spacja – puste miejsce, po którym można się poruszać.

Maksymalny rozmiar labiryntu to 1024x1024. Plik może również przyjąć formę binarną, której postać jest szczegółowo opisana w osobnym dokumencie. W tym przypadku, zostanie on najpierw przekształcony do postaci tekstowej. Labirynt wraz z zaznaczoną ścieżką zostają zapisane w pliku lab.txt, a na konsoli zostają wyświetlone dane labiryntu i ścieżka przejścia.

2 Ustalenia wstępne

Język Programowania	C
Format Pobierania Danych	TXT, BIN
Sposób wersjonowania i współpracy	GitHub

3 Podział plików

3.1 Pliki główne

- labirynt.c - główny plik programu, znajduje się w nim funkcja main.c. Jest on odpowiedzialny za wybór trybu pracy programu (dla pliku tekstowego, lub binarnego), w zależności od argumentów wejściowych, przechowuje podstawowe dane dotyczące labiryntu, oraz w przypadku plików binarnych zajmuje się ich odekodowaniem.
- dane_labiryntu.c - wyszukuje potrzebne dane dotyczące labiryntu, takich jak jego wymiary, oraz położenie początku i końca
- oznaczenie.c - zajmuje się przechodzeniem przez labirynt, odnajdywaniem ścieżki, oraz zapisywaniem jej do pliku wyjściowego lab.txt
- wyświetlacz.c - zajmuje się wyświetlaniem na konsoli kolejno: danych labiryntu, Labiryntu w postaci tekstowej, Labiryntu z zaznaczoną ścieżką i opisem krok po kroku kolejnych przejść, np. FORWARD 3, UP 2 itd.

3.2 Pliki testowe

- smallmaze.txt, maze.txt, bigmaze.txt, hugemaze.txt - coraz większych rozmiarów pliki labiryntów (od 21x21 do 1024x1024); maze.txt jest domyślnie wywoływany plikiem przez program, w przypadku braku argumentu wejściowego
- maze20%.txt - plik labiryntu normalnego rozmiaru z usuniętymi ścianami w liczbie 20/
- maze.bin - duży plik z labiryntem (512x512) zakodowany w postaci binarnej

4 Implementacja odnajdywania ścieżki

- Do znalezienia ścieżki w labiryncie wykorzystany został algorytm Depth First Search, który wchodzi coraz głębiej rekurencyjnie w labirynt i cofa się dopiero, gdy napotka przeszkodę.
- Do pliku lab.txt skopiowany zostaje labirynt z pliku tekstowego wejściowego lub otrzymanego z pliku binarnego, funkcja oznacz w pliku oznaczanie.c zapisuje 1 na każdym polu na które trafi, a w przypadku napotkania przeszkody lub trafienia na pole z wartością 1, cofa się usuwając po drodze jedynki z pól, aż napotka na rozgałęzienie mające drogę, po której nie szedł.
- Po znalezieniu ścieżki, funkcja PoprawneOznaczenie przechodzi ponownie po pliku, tym razem jednak zamieniając wartości 1 w polach na pierwsze litery kierunków drogi (np. d d d l l g g r ...)