Labirynt

Borys Piątek, Mateusz Ciurzyński

Marzec 2024

Spis treści

1 Wstęp

1.1 Cel projektu

Celem projektu było stworzenie programu w języku C umożliwiającego znalezienie ścieżki w uprzednio wygenerowanym labiryncie podanym jako plik tekstowy zgodnie z założeniami.

1.2 Założenia

Poniżej podane są kryteria i założenia dotyczące programu:

- Maksymalny rozmiar pliku labiryntu to 2049 x 2049 znaków, gdzie maksymalny realny rozmiar labiryntu, czyli pól po których można przejść, to 1024 x 1024. Plik składa się z znaków 'X', ' ', 'P' i 'K'. Znak 'X' oznacza ścianę, znak ' ' oznacza miejsce, po którym można się poruszać, znak 'P' oznacza początek labiryntu, a znak 'K' koniec labiryntu.
- Minimalnie rozmiar labiryntu to 3 x 3 (znaki).
- Labirynt zawiera jedno wejście i jedno wyjście.
- Poszczególne poziomy labiryntu oddzielane są przy pomocy znaku nowej linii.
- Maksymalna pamięć zużywana w czasie całego swojego działania to 512 kB.
- Znaleziona ma być jedna ścieżka od wejścia do wyjścia.

2 Specyfikacja funkcjonalna

2.1 Wczytywanie labiryntu

Wczytywanie labiryntu jest realizowane poprzez przesuwanie pozycji kursora w pliku tekstowym i pobieranie na raz wyłącznie jednego znaku wskazanego przez kursor. Sposób przesuwania kursora symuluje działanie tablicy dwuwymiarowej - przesuwamy się po dwóch współrzędnych odpowiadających wierszom i kolumnom w pliku tekstowym. Wybór takiej metody pozwala na znaczne ograniczenie pamięci używanej przez program w kontekście pobierania danych o planszy labiryntu. W miarę przechodzenia labiryntu przez algorytm przeszukujący, pobierane są odpowiednio kolejne znaki, ale w jednym momencie zapisany jest w dalszym ciągu tylko jeden znak.

2.2 Algorytm przeszukiwania labiryntu

Szukanie ścieżki rozwiązującej labirynt jest realizowane przy użyciu algorytmu DFS. Samo przechodzenie labiryntu opiera się na pobieraniu znaków zgodnie z opisaną metodą i wykonywaniu odpowiednich czynności, ale inne niezbędne dane wytwarzane przez algorytm zapisywane są do dwóch nowych plików tekstowych (tworzonych przez program). Jeden z nich zawiera informacje potrzebne do poprawnego bieżącego działania programu, natomiast drugi plik zawiera wynik działania całego programu - ścieżkę od początku do końca labiryntu.

3 Kompilacja i wywołanie programu

3.1 Kompilacja

Kompilacja programu odbywa się przy użyciu pliku "Makefile", wpisując polecenie "make". Tworzy to plik wykonwywalny "main.out". Polecenie "make clean" umożliwia usunięcie plików wykonawczych.

3.2 Wywołanie

Aby wywołać program należy użyć polecenia "./main.out [plik]". Argumentem potrzebnym do wywołania jest ścieżka do pliku labiryntu. Dodatkowo można zawrzeć parametry opcjonalne oddzielone spacją w formacie:
" -<oznaczenie literowe parametru> ".
Przykład wywołania:
"./main.out maze"

3.3 Parametry opcjonalne

Program umożliwia podanie opcjonalnych parametrów podczas wywołania, które mogą zmodyfikować jego sposób działania lub wypisać dodatkowe informacje:

- "-h" - wyświetla krótki opis programu oraz sposób jego użycia, np. pozostałe parametry

4 Sytuacje wyjątkowe

Program zabezpieczony jest przed nieprawidłowym wywołaniem oraz przed wywołaniem z nieprawidłowym plikiem. W przypadku błędów program wyświetli odpowiedni komunikat.