

Autor: Mateusz Buczyński

## ***Projekt: Algorytm generujący labirynt***

### **Opis programu:**

Po wystartowaniu programu musimy wpisać wymiary labiryntu jaki ma zostać wygenerowany, wysokość oraz szerokość. Następnie program przenosi nas do menu gdzie wybieramy gdzie ma zostać zapisany lub wyświetlony wygenerowany labirynt.

*Dla szybkiego przetestowania labiryntu w menu znajduje się opcja wygenerowania kilku labiryntów (w tym celu należy w menu wybrać K)*

Jeśli wybraliśmy opcję S czyli zapisanie naszego labiryntu do pliku, należy podać dodatkowo nazwę pliku. D wyświetla labirynt bezpośrednio na ekranie. E zamyka program (jeśli nie zostanie wybrana opcja zamknięcia programu, po wykonaniu zadania będziemy mogli wpisać znów rozmiar labiryntu i wykonać działania ponownie).

### **Opis kodu:**

Za wypisanie rezultatu programu odpowiada `''.join(a + ['\n'] + b` zawarte w pętli `for`. Co więc jeśli wypisalibyśmy nasz labirynt na ekran po zadeklarowaniu `vis`, `ver`, `hor`? Dostalibyśmy nasz labirynt bez wydrążonych w nim korytarzy.

Funkcja `walk` inicjalizuje drażenie labiryntu w losowym miejscu w labiryncie. Każdy punkt jest odwiedzony dokładnie raz. Odwiedzone miejsca są zapisywane w `vis`. Funkcje `walk` wywołujemy rekurencyjnie.

Poszczególne `if` są odpowiedzialne za usuwanie ścian w naszym labiryncie.

Pozycje naszego kursora przechowujemy w zmiennej `d` (położenie sąsiadów naszego punktu).

Funkcja `shuffle` z biblioteki `random` losuje permutacje sekwencji `d`. Funkcja `randrange` losuje liczby pseudolosowe w jakimś konkretnym zakresie, w tym przypadku od 0 do podanych rozmiarów.

## **Algorytm:**

1. Wybieramy losowo pozycję startową i ustawiamy ją jako komórkę bieżącą..

2. Badamy sąsiednie komórki i zapisujemy do listy tylko te, które posiadają 4 ściany (czyli są nieodwiedzone).

3. Jeśli lista nie jest pusta, to wybieramy jedną z nich w sposób losowy i usuwamy ścianę pomiędzy wylosowaną komórką, a tą którą obecnie się zajmujemy. Następnie przechodzimy do wylosowanej komórki (staje się ona bieżącą komórką).

4. Jeżeli wszystkie komórki sąsiadujące z bieżącą komórką zostały odwiedzone, to cofamy się o komórkę i ustawiamy ją jako bieżącą.

5. Należy powtarzać kroki 2-4, aż do momentu odwiedzenia wszystkich komórek.