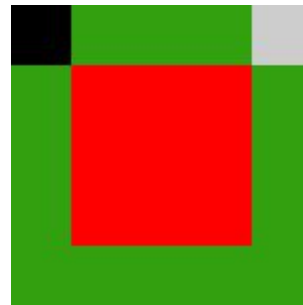


W [plikach CSV](#) zapisane są cztery mapy labiryntów różnej wielkości. Labirynt składa się z:

- ścieżek o szerokości jednego znaku oznaczonych zerami,
- ścian oznaczonych jedynkami,
- punktu początkowego oznaczonego cyfrą 2,
- punktu końcowego oznaczonego cyfrą 3.

Przykładowy labirynt (*map\_1.csv*) wraz z jego wizualizacją zaprezentowany jest poniżej:

```
2,0,0,0,3
0,1,1,1,0
0,1,1,1,0
0,1,1,1,0
0,0,0,0,0
```



Kolory w wizualizacji mają następujące znaczenie:

- czarny - punkt początkowy,
- szary - punkt końcowy,
- czerwony - ściany,
- zielony - dozwolone ścieżki.

Zadanie polega na napisaniu programu znajdującego najkrótszą trasę od punktu początkowego do punktu końcowego poruszając się jedynie po dozwolonych ścieżkach. Można założyć, że istnieje tylko jedna najkrótsza ścieżka. Ruch nie może odbywać się po przekątnych (jedynie dopuszczalne kierunki to lewo, prawo, góra i dół).

Program powinien przez standardowe wejście przyjmować dwie ścieżki w podanej kolejności:

1. ścieżkę do pliku CSV z mapą,
2. ścieżkę do wyjściowego (nieistniejącego) pliku CSV, który musi utworzyć.

Wyjściowy plik CSV powinien zawierać jedynie pary współrzędnych punktów tworzących najkrótszą trasę wraz z punktem początkowym i końcowym. Kolejność współrzędnych w parze powinna być następująca: współrzędna związana z wysokością, a następnie współrzędna związana z szerokością. Zakładamy, że układ współrzędnych zaczyna się w lewym górnym rogu.

Prawidłowa zawartość wyjściowego pliku CSV dla powyższego przykładu wygląda następująco:

```
0,0
0,1
0,2
0,3
0,4
```