

Zadanie: H

Labirynt



Autor: Szymon Wąsik, Dostępna pamięć: 64 MB

3 czerwca 2006

Głęboko w podziemiach Bajtocji znajduje się zapomniany labirynt pełen niebezpiecznych pułapek. Niedawno odważny rycerz Bajtazar postanowił przebyć labirynt, ponieważ na pierwszą osobę która tego dokona czekają niewyobrażalne skarby. Jednak aby jak najmniej narażać swoje życie, chciałby przebyć labirynt jak najkrótszą drogą.

Labirynt składa się z kwadratowych pól. Z każdego pola można przejść na cztery sąsiadujące z nim pola, oczywiście jeżeli nie znajduje się na nich ściana. Dodatkowym utrudnieniem są strażnicy, którzy znajdują się na niektórych polach. Każdy strażnik spogląda w jeden z czterech kierunków (na wschód, zachód, północ lub południe). Na pole na którym znajduje się strażnik można wejść tylko z kierunku, w który spogląda strażnik lub z do niego przeciwnego (np. dla południa jest to północ). Jednak w momencie gdy wchodzimy na pole z kierunku, w którym spogląda strażnik, wymaga on, abyśmy podali mu odpowiedni klucz. Jest wiele różnych rodzajów kluczy i wielu strażników może wymagać tego samego typu klucza. W momencie gdy Bajtazar wchodzi w ten sposób na pole, na którym znajduje się strażnik, traci wymagany przez niego klucz. Aby otrzymać klucz Bajtazar musi wejść na zajmowane przez strażnika pole z kierunku przeciwnego do tego, w którym patrzy strażnik. Wtedy zaskoczony strażnik oddaje Bajtazarowi jeden klucz. Pojawia się jednak mały problem. Jeżeli Bajtazar znajdzie od tyłu strażnika, posiadając już klucz jaki mógłby dostać, strażnik wpada w furię i go zabija, co oczywiście nie powinno nastąpić. W momencie gdy Bajtazar rozpoczyna swoją podróż, nie posiada żadnego klucza.

Zadanie

Napisz program który:

- wczyta ze standardowego wejścia opis labiryntu,
- wyliczy jaka jest najmniejsza ilość pól, które musi przejść Bajtazar, aby przedostać się z wejścia labiryntu do jego wyjścia,
- wypisze wynik na standardowe wyjście.

Wejście

W pierwszej linii wejścia znajduje się liczba d określająca ilość przypadków testowych. Następnie znajduje się opis kolejnych przypadków. W pierwszej linii znajduje się zawsze sześć liczb całkowitych $1 \leq l_x, l_y, s_x, s_y, k_x, k_y \leq 270$, $1 \leq s_x, k_x \leq l_x$, $1 \leq s_y, k_y \leq l_y$ oznaczające odpowiednio wymiary labiryntu (wysokość i szerokość) oraz współrzędne wejścia do labiryntu i jego wyjścia. Następnie znajduje się l_x wierszy, w każdym z nich znajduje się l_y znaków 0 lub 1. Zero oznacza, że pole jest wolne, natomiast jeden, że znajduje się na nim ściana. Następnie znajduje się jedna liczba k oznaczająca liczbę występujących strażników. W kolejnych k wierszach znajduje się opis strażników. Każdy strażnik opisany jest za pomocą trzech liczb s_{x_i} , s_{y_i} , k_{l_i} oraz litery dir_i , $1 \leq s_{x_i} \leq l_x$, $1 \leq s_{y_i} \leq l_y$, $1 \leq k_{l_i} \leq 8$ oznaczające odpowiednio pole na którym znajduje się strażnik, rodzaj

obsługiwane przez niego klucza oraz kierunek w który strażnik jest zwrócony (S - w dół, N - w górę, E - w prawo, W - w lewo). Na każdym polu znajduje się co najwyżej jeden strażnik.

Wyjście

Dla każdego przypadku testowego program powinien wypisać dokładnie jedną linię zawierającą jedną liczbę - ilość pól, które musi pokonać Bajtazar, aby przedostać się z pola startowego do końcowego (nie licząc pola startowego). Możesz założyć, że taka droga zawsze istnieje i jest krótsza niż 5000 pól.

Przykład

Dla danych wejściowych:

```
1
7 8 6 3 6 6
11111111
10010001
10010101
10010001
10000011
11011011
11111111
5
3 3 1 S
4 3 2 S
5 4 1 W
5 6 2 S
3 7 2 N
```

poprawnym wynikiem jest:

```
21
```

Optymalna trasa jest następująca: N, W, N, N, N, E, S (dostajemy klucz 1), S (dostajemy klucz 2), S, E (tracimy klucz 1), E, N, N, N, E, E, S (pozbywamy się klucza 2), S, W, S (dostajemy klucz 2), S.