

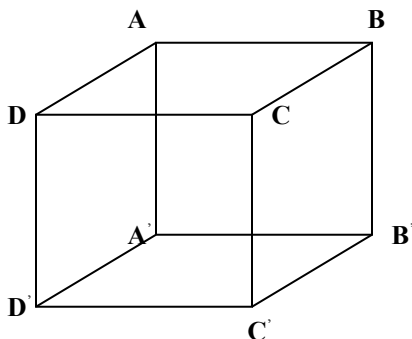
Cub

100 puncte

Fișiere sursă: **cub.c**, **cub.cpp** sau **cub.pas**

Stația de cercetări astronomice ONI–2006 are forma unui cub cu latura de lungime N unități. Fiecare dintre cele 6 fețe ale stației este împărțită în $N \times N$ pătrate cu latura unitate. Cosmonauții însărcinați cu întreținerea stației se pot deplasa pe suprafața cubului, folosind șine speciale plasate în unele dintre pătrățelele unitate, pentru a efectua diverse reparații. Unele pătrate unitate, dintre cele prevăzute cu șine, conțin trape de intrare în interiorul stației.

Din cauza radiațiilor cosmice nocive, cosmonauții trebuie să petreacă un timp cât mai scurt în exteriorul stației astfel încât, după terminarea unei reparații, aceștia trebuie să se reîntoarcă în interiorul stației utilizând cea mai apropiată trapă de acces. Cosmonautul se află inițial într-un pătrățel prevăzut cu șine. El se poate deplasa dintr-un pătrățel ce conține șine, în altul, aflat în vecinătatea poziției curente (pe orizontală sau verticală), pătrățel prevăzut de asemenea cu șine. Cosmonautul se poate deplasa și de pe o față a cubului pe o față vecină, traversând latura comună. Fețele cubului sunt descrise conform figurii de mai jos.



- Fața 1: **ABCD** - elementul (1,1) în A și elementul (N,N) în C
- Fața 2: **DCC'D'** - elementul (1,1) în D și elementul (N,N) în C'
- Fața 3: **B'A'D'C'** - elementul (1,1) în B' și elementul (N,N) în D'
- Fața 4: **BAA'B'** - elementul (1,1) în B și elementul (N,N) în A'
- Fața 5: **CBB'C'** - elementul (1,1) în C și elementul (N,N) în B'
- Fața 6: **ADD'A'** - elementul (1,1) în A și elementul (N,N) în D'

Cerință

Scrieți un program care să determine distanța minimă de la poziția inițială a cosmonautului până la o trapă de acces, precum și numărul trapelor aflate la distanță minimă.

Date de intrare

Datele de intrare se citesc din fișierul **cub.in**, care are următoarea structură:

Pe prima linie se află două numere naturale N și K , separate printr-un spațiu, reprezentând lungimea laturii cubului, respectiv numărul trapelor de acces.

Pe a doua linie avem numerele naturale F , L , C , separate prin spațiu, desemnând pătratul unitate pe care se află inițial cosmonautul, unde $F \in \{1,2,3,4,5,6\}$ reprezintă numărul unei fețe a cubului, iar L și C reprezintă coordonatele poziției inițiale a cosmonautului, relative la colțul (1,1) al feței cu numărul F (L – numărul liniei, C – numărul coloanei)

Pe următoarele K linii se află câte 3 numere naturale, separate prin spațiu, reprezentând coordonatele celor K trape de acces (numărul feței, numărul liniei, respectiv numărul coloanei).

Pe următoarele $6N$ linii se află cele 6 matrice pătratice care descriu fețele cubului, având elemente din mulțimea $\{0, 1\}$ (0 – șină de acces, 1- pătrat inaccesibil). Elementele unei fețe sunt date pe linii, de la (1,1) până la (N,N).

Date de ieșire

Rezultatele se scriu în fișierul **cub.out**, care are următoarea structură:

Pe prima linie se va scrie numărul natural LG , reprezentând lungimea drumului minim parcurs de cosmonaut. Lungimea drumului este dată de numărul pătratelor unitate parcurse, inclusiv poziția inițială și poziția finală. Pe a doua linie se va afișa numărul natural T , reprezentând numărul trapelor de acces aflate la distanță minimă față de poziția inițială a cosmonautului.



Ziua 1

Olimpiada Națională de Informatică
Târgoviște, 14-22 aprilie 2006

Clasa a X-a

Restricții și precizări

$1 \leq N \leq 50$

$1 \leq F \leq 6$

$1 \leq L, C \leq N$

Cosmonautul se află inițial într-o poziție accesibilă (prevăzută cu șină).

Există cel puțin o trapă accesibilă din poziția inițială a cosmonautului.

Exemplu:

cub.in	cub.out	explicații
3 2 2 2 3 5 2 2 3 2 2 1 1 1 1 0 0 1 0 1 0 0 1 0 1 0 0 0 0 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1	12 1	Traseul ilustrat are costul minim 12 și unește elementul (2,3) de pe fața 2 cu elementul (2,2) de pe fața 5. Cealaltă trapă de acces se află pe fața 3, într-o poziție inaccesibilă, deci numărul trapelor aflate la distanța minimă față de poziția de start este 1. 

Timp de executare/test: 0.5 secunde pentru Windows

0.2 secunde pentru Linux