**NGÂN HÀNG CÂU HỎI THI THỰC HÀNH PHÒNG MÁY**

**Môn:** Cấu trúc dữ liệu và giải thuật. **Phần:** Hàng đợi

**MỤC LỤC**

[**BÀI 1. DSA\_P006. QUÂN VUA TRÊN BÀN CỜ** 1](#_Toc206334438)

[**BÀI 2. DSA\_P058. SỐ BƯỚC DI CHUYỂN ÍT NHẤT** 1](#_Toc206334439)

[**BÀI 3. DSA\_P069. VUA VÀ QUÂN MÃ** 2](#_Toc206334440)

[**BÀI 4. DSA\_P070. THÍ NGHIỆM VẬT LÝ** 3](#_Toc206334441)

[**BÀI 5. DSA\_P097. DI CHUYỂN ROBOT TRÊN BẢN ĐỒ LỚN** 3](#_Toc206334442)

[**BÀI 6. DSA\_P133. DI CHUYỂN ROBOT** 4](#_Toc206334443)

[**BÀI 7. DSA\_P135. DI CHUYỂN 4 QUÂN CỜ** 7](#_Toc206334444)

# **BÀI 1. DSA\_P006. QUÂN VUA TRÊN BÀN CỜ**

Trên bàn cờ vua kích thước 8\*8 thì quân vua được phép di chuyển đến cả 8 ô liền kề theo cả đường dọc và đường chéo. Tất nhiên quân vua sẽ không thể di chuyển được ra ngoài bàn cờ. Cho 2 ô trên bàn cờ gọi là ô bắt đầu và ô kết thúc. Hãy tính xem quân vua cần ít nhất bao nhiêu bước để di chuyển từ ô bắt đầu đến ô kết thúc.

**Input.** Có 2 cặp số nguyên s1, s2 và f1, f2 lần lượt và vị trí ô bắt đầu và ô kết thúc. Các vị trí đảm bảo nằm trong phạm vi bàn cờ.

**Output**. Số bước đi ít nhất của quân vua

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 3 1 6 | 3 |
| 5 5 5 6 | 1 |

# **BÀI 2. DSA\_P058. SỐ BƯỚC DI CHUYỂN ÍT NHẤT**

Cho ma trận A kích thước N\*M.

Hãy tìm số bước đi ít nhất để di chuyển từ vị trí A[1][1] đến vị trí A[N][M].

Biết rằng mỗi bước từ vị trí (i, j) ta có thể di chuyển theo một trong ba hướng:

* Hướng xuống dưới với số ô di chuyển là hiệu hai giá trị A[i][j] và A[i+1][j]
* Hướng sang phải với số ô di chuyển là hiệu hai giá trị A[i][j] và A[i][j+1]
* Hướng chéo xuống với số ô di chuyển là hiệu của hai giá trị A[i][j] và A[i+1][j+1]

**Input:**

* Dòng đầu tiên đưa vào số lượng test T.
* Dòng tiếp theo đưa vào các bộ test. Mỗi bộ test gồm hai phần:

Phần thứ nhất là hai số N, M;

Phần thứ hai là các phần tử của ma trận A[][]; các số được viết cách nhau một vài khoảng trống.

* T, N, M, A[i][j] thỏa mãn ràng buộc: 1 ≤ T ≤ 100; 1 ≤ N, M, A[i][j] ≤ 103 .

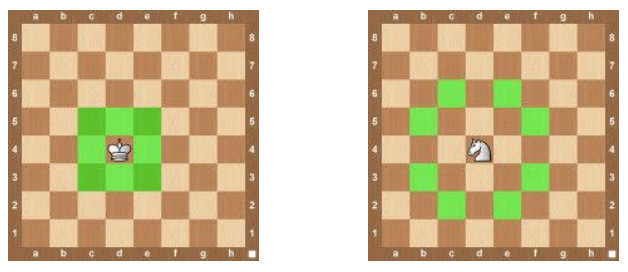
**Output:** Đưa ra kết quả mỗi test theo từng dòng. Nếu không tìm được đường đi ghi ra -1

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 1  3 3  2 1 2  1 2 4  1 3 2 | 3 |

# **BÀI 3. DSA\_P069. VUA VÀ QUÂN MÃ**

Vua Titan và các hiệp sĩ bàn tròn thường gặp nhau vào đầu năm mới để ăn mừng tình bạn của họ. Để tưởng nhớ sự kiện này, chúng ta xem xét một trò chơi, trong đó có một quân vua và một quân mã được đặt ngẫu nhiên trên 2 ô riêng biệt. Bàn cờ có kích thước N×N, trên bàn cờ có một số ô cấm, những ô còn lại là những ô tự do – ô có thể di chuyển vào được. 2 ô đặt quân mã và quân vua đang đứng ở các ô tự do.



Tại mỗi bước cả 2 quân đều phải di chuyển theo quy tắc và không được đi vào ô cấm, hãy tìm cách di chuyển để chúng gặp nhau nhanh nhất.

**Input:**

* Dòng đầu tiên là số nguyên N (N ≤ 100).
* N dòng tiếp theo, mỗi dòng gồm 1 xâu có N ký tự mô tả bàn cờ. Trong đó:
* “.” thể hiện ô trống
* “#” thể hiện ô cấm
* “T” thể hiện vị trí vua đang đứng
* “M” thể hiện vị trí quân mã đang đứng.

**Output.** In ra đáp án là số bước ít nhất để các quân gặp nhau. Nếu không thể gặp nhau, in -1.

**Ví dụ.**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 5  M….  …..  .#...  .#..#  …#T | 2 |

# **BÀI 4. DSA\_P070. THÍ NGHIỆM VẬT LÝ**

Có một chiếc đèn laze tại vị trí (xA, yA) và bạn cần phải chiếu sáng vị trí (xB, yB). Trên hệ thống mô phỏng có N vị trí đã gắn sẵn giá gương, bạn được phép lựa chọn có đặt gương vào các vị trí này hay không. Nếu không có gương, tia laze sẽ tiếp tục truyền thẳng theo hướng song song với trục Ox hoặc trục Oy. Nếu bạn đặt gương, ánh sáng laze sẽ bị bẻ vuông 90 độ (sang trái, hoặc phải).

Nhiệm vụ của bạn là hãy xác định số lượng gương ít nhất cần sử dụng, sao cho ánh sáng laze có thể chiếu sáng tới được vị trí B.

**Input**

* Dòng đầu tiên là số nguyên N (1 ≤ N ≤ 105) và xA, yA, xB, yB.
* N dòng tiếp theo, mỗi dòng gồm tọa độ của các vị trí đặt giá gương. (Vị trí đặt gương có thể trùng với A hoặc B, và không có vị trí nào đặt được 2 gương).
* Các tọa độ nằm trong phạm vi từ 0 đến 109.

**Output**

In ra một số nguyên là đáp án của bài toán. Nếu vị trí B không thể được chiếu sáng, in ra -1.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 4 0 0 9 3  1 7  0 3  3 0  3 3 | 1 |

**Giải thích test:** Đặt gương tại vị trí (0, 3).

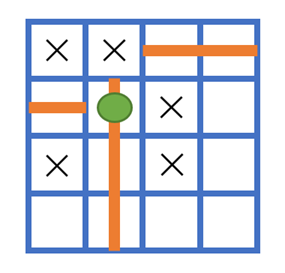
**Giới hạn thời gian:** 2s

**Giới hạn bộ nhớ:** 200000 Kb

# **BÀI 5. DSA\_P097. DI CHUYỂN ROBOT TRÊN BẢN ĐỒ LỚN**

Trên sân thi đấu được có giới hạn 109x 109, các chú robot sử dụng công nghệ dò đường line sẽ di chuyển theo các vạch chỉ đường có sẵn. Mỗi bước, robot sẽ dịch chuyển được 1 ô đơn vị theo một trong 8 hướng. Robot có thể thực hiện được phép quay 45 độ hay 90 độ nếu như ô kế cận cũng đã được kẻ vạch.

Hình vẽ dưới đây minh họa các hướng di chuyển của robot trên một sân thi đấu có sẵn, các vị trí đánh dấu X mô tả robot không được đi theo hướng này.



Nhiệm vụ của bạn là hãy lập chương trình để di chuyển robot vị trí (xA, yA) tới vị trí (xB, yB) với số bước ít nhất. Input đảm bảo hai vị trí A và B chắc chắn đã được kẻ vạch.

**Input:**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 10).
* Mỗi test gồm số nguyên xA, yA, xB, yB mô tả tọa độ hai điểm A và B.
* Tiếp theo là số nguyên N (N ≤ 105), mô tả số vạch đường chỉ dẫn.
* N dòng tiếp theo, mỗi dòng gồm 3 số nguyên x, y1, y2 (y1 ≤ y2) cho biết có một vạch chỉ dẫn từ ô (x, y1) tới (x, y2).
* Input đảm bảo tổng số lượng số ô được kẻ vạch trên sân thi đấu không vượt quá 105.

**Giới hạn**. 0 ≤ x, y1, y2 ≤ 109.

**Output.** Với mỗi test, in ra số bước di chuyển ít nhất tìm được.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  0 6 2 2  3  0 6 6  1 6 10  2 2 5  1 1 2 10  2  1 1 4  2 8 10 | 5  -1 |

# **BÀI 6. DSA\_P133. DI CHUYỂN ROBOT**

Quy định của cuộc thi Robocon năm 2022 như sau:

* Các robot sẽ thi đấu trên một bản đồ gồm N ô (đỉnh) vị trí khác nhau được đánh số từ 1 đến N và M đoạn đường một chiều nối các cặp đỉnh.
* Mỗi ô được tô bằng một trong ba màu 1, 2, 3.
* Mỗi đoạn đường cũng được tô bằng một trong ba màu 1, 2, 3 nhưng sẽ bị biến đổi màu theo thời gian. Cụ thể, nếu ở thời điểm t, đoạn đường có màu là c thì ở thời điểm t + 1 đoạn đường sẽ có màu là (c mod 3)+1.
* Mỗi đội chơi phải điều khiển hai robot cùng di chuyển trên bản đồ để hai robot gặp nhau sớm nhất tại một ô. Ban đầu (tại thời điểm 0), robot thứ nhất đứng ở ô 1, robot thứ hai đứng ở ô 2, quy tắc di chuyển của hai robot là:
* Mỗi đơn vị thời gian, cả hai robot cùng phải di chuyển qua một đoạn đường;
* Giả sử ở thời điểm t robot thứ nhất đang đứng ở ô i, robot thứ hai đang đứng ở ô j, khi đó:

+ Robot thứ nhất buộc phải di chuyển theo một trong các đoạn đường có màu ở thời điểm t cùng màu với màu ở ô j,

+ Robot thứ hai buộc phải di chuyển theo một trong các đoạn đường có màu ở thời điểm t cùng màu với màu ở ô i.

Cho thông tin về bản đồ với màu ban đầu của các đoạn đường, hãy xác định thời điểm hai robot gặp nhau sớm nhất.

**Input**

* Dòng đầu tiên là số lượng bộ test T (T ≤ 10).
* Mỗi test bắt đầu gồm M + 2 dòng, trong đó:
* Dòng thứ nhất là 2 số nguyên N và M (M ≤ ).
* Dòng thứ hai ghi N số nguyên, là màu của ô thứ i.
* M dòng tiếp, mỗi dòng gồm 3 số nguyên u, v, c mô tả đoạn đường từ u đến v có màu tại thời điểm 0 là c.

**Subtask:**

* **Subtask 1:** 2 ≤ N ≤ 10
* **Subtask 2:** 2 ≤ N ≤ 50
* **Subtask 3:** 2 ≤ N ≤ 200

**Output**. Với mỗi test, hãy in ra đáp án trên một dòng. Nếu không tìm được đáp án, in ra -1.

**Ví dụ:**

|  |  |
| --- | --- |
| **Input** | **Output** |
| 2  3 4  1 2 3  1 2 2  2 1 2  2 3 1  3 1 1  3 4  1 2 3  1 2 3  2 1 2  2 3 1  3 1 1 | 2  -1 |

**Giải thích test:**

* **Test 1:**

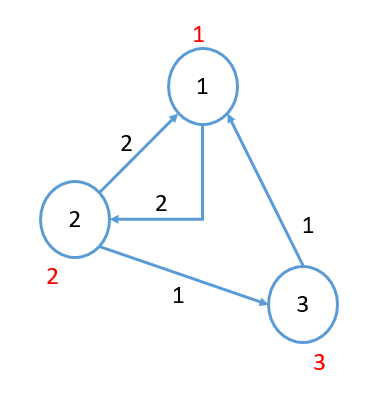
t = 0, 1 🡪 2 và 2 🡪 3 (Cạnh 1 🡪 2 có màu 2, cạnh 2 🡪3 có màu 1)

t = 1, 2 🡪 1 và 3 🡪 1 (Cạnh 2 🡪 1 có màu 3, cạnh 3 🡪1 có màu 2)

t = 2, 2 robot gặp nhau tại đỉnh 1.

* **Test 2:**

Tại t = 0, đỉnh 1 không có cạnh nào có màu bằng 2, do đó cần in ra -1.



**Giới hạn thời gian:** 2s

**Giới hạn bộ nhớ:** 500000 Kb

# **BÀI 7. DSA\_P135. DI CHUYỂN 4 QUÂN CỜ**

