

TỔNG QUAN:

1. Tất cả các bài tập đều có dữ liệu vào/ra ở tệp.
2. Phía sau tên của mỗi bài đều có quy định <tên bài>.* Trong đó tên bài là tên chung cho tên tệp chương trình, tệp input, tệp output, dấu * là CPP hoặc PAS với tên chương trình, INP với tệp input, OUT với tệp output.
3. Có 5 bài, thời gian làm bài từ 13h40 đến 15h20.
4. Chấm bài lúc 15h21, chỉ chấm duy nhất 1 lần.

II. LẬP TRÌNH GIẢI CÁC BÀI TOÁN SAU

Bài 1: 3SUM (3sum.*)

Bài toán này yêu cầu kỹ năng lập trình code_not_bug.

Dữ liệu vào:

- + Dòng đầu tiên gồm một số nguyên dương T ($1 \leq T \leq 200$).
- + T dòng tiếp theo, dòng thứ i gồm một số nguyên dương N_i ($1 \leq N_i \leq 10000$).

Kết quả: Gồm T dòng, dòng thứ i gồm ba số nguyên dương lần lượt là tổng của N_i số nguyên dương đầu tiên, tổng của N_i số nguyên dương lẻ đầu tiên và tổng của N_i số nguyên dương chẵn đầu tiên.

Ví dụ:

Input	Output
3	1 1 2
1	55 100 110
10	501501 1002001 1003002
1001	

Bài 2: HỒ NƯỚC (lake.*)

Cánh đồng của nông dân John bị ngập lụt! Có thể mô tả cánh đồng như một bảng kích thước M dòng \times N cột ($0 < N, M \leq 100$), trong đó một số ô bị ngập trong nước và một số ô khô ráo. Có K ô bị ngập ($0 < K \leq M \times N$), được mô tả bằng các cặp số (x_k, y_k) là tọa độ của chúng. Các tọa độ đôi một phân biệt.

Một tập các ô được gọi là hồ nếu như tất cả các ô trong tập đều bị ngập nước và có thể di chuyển từ 1 ô sang ô khác cùng trong tập bằng cách đi qua ô ngập nước kề cạnh. Xác định kích thước của hồ lớn nhất (nhiều ô nhất) trên cánh đồng.

Dữ liệu vào:

- + Dòng 1: ghi lần lượt 3 số nguyên dương M, N, K .
- + Dòng 2 ... $K + 1$: Mỗi dòng chứa 2 số nguyên dương x_i, y_i là tọa độ của ô ngập nước thứ i .

Kết quả: 1 số duy nhất là kích thước lớn nhất của hồ nước có trong cánh đồng.

Ví dụ:

Input	Output
3 4 5	4
3 2	
2 2	
3 1	
2 3	

Bài 3: JUSTIN BIEBER (justinbieber.*)

Một em bé đang cố gắng giải bài toán 8 quân hậu kinh điển: Đặt 8 quân hậu trên bàn cờ vua kích thước 8×8 ô vuông sao cho không có bất kì hai quân hậu nào nằm trên cùng một cột, hàng, hoặc đường chéo (nói các khác, hai quân hậu ấy không thể “ăn” nhau). Em bé hiểu khái niệm hàng và cột rất rõ song đường chéo thì còn nhiều khúc mắc. Em đã thành công với việc đặt 8 quân hậu sao cho không có hai quân hậu nào nằm trên cùng hàng hoặc cột nhưng vẫn có khả năng một vài quân hậu đang đang nằm trên một đường chéo.

Sau khi em sắp xếp xong, em đã đưa kết quả này cho các bạn đội (dự) tuyển. Ngoài ra, BTC của cuộc thi còn cung cấp cho các thí sinh đáp án của bài toán quân hậu trên. Em bé nhờ các bạn rằng, từ bàn cờ mà em vừa sắp xếp, hãy thực hiện mtotoj số ít nhất các di chuyển quân hậu để đưa về đáp án mà các bạn đã được cung cấp.

Biết rằng từ kết của của em bé, có thể di chuyển các quân hậu để trở thành kết quả đúng. Giả sử trong một bước, một quân hậu có thể đi đúng 1 ô dọc theo hàng hoặc cột để đến một ô trống khác trong bàn cờ.

Hãy giúp em bé trong trường hợp tổng quát bàn cờ kích thước $N \times N$.

Dữ liệu vào:

+ Có nhiều test trong cùng một Input.

+ Mỗi test bắt đầu bằng số N ($3 < N < 17$). Tiếp theo đó là hai dòng, mỗi dòng có N số. Dòng thứ nhất gồm các số a_1, a_2, \dots, a_N cho biết cách xếp quân hậu thứ i của em bé được đặt ở cột a_i , hàng thứ i . Tương tự, dòng thứ 2 gồm các số b_1, b_2, \dots, b_N cho biết cách xếp quân hậu đúng mà bạn cần biến đổi bàn cờ ban đầu đến trạng thái đó.

+ Kết thúc Input là một số 0.

Kết quả: Với mỗi test, in ra một số duy nhất trên một dòng là số bước di chuyển ít nhất của các quân hậu.

Ví dụ:

Input	Output
4	6
1 2 3 4	2
3 1 4 2	0
4	8
3 2 4 1	

3 1 4 2	
5	
5 3 1 4 2	
5 3 1 4 2	
5	
1 5 2 4 3	
3 1 4 2 5	
0	

Bài 4: TRIPLE (triple.*)

Trò chơi “Triple” (Bộ ba) được diễn ra trên một bảng kích thước $N \times N$. Mirko và Slavko đầu tiên viết một số chữ cái lên một số ô của bảng. Không có loại chữ cái nào được viết nhiều hơn một lần.

Sau đó hai người đếm số bộ ba chữ cái thẳng hàng (một bộ ba được gọi là thẳng hàng nếu tồn tại một đường thẳng đi qua tâm của cả ba ô đó).

Hãy xác định số bộ ba thẳng hàng.

Dữ liệu vào:

+ Dòng thứ nhất chứa một số nguyên N ($1 \leq N \leq 100$).

+ N dòng tiếp theo, mỗi dòng là một xâu N ký tự, mỗi ký tự là một chữ cái in hoa hoặc ‘.’ ứng với ô trống.

Kết quả: Một số nguyên duy nhất là số bộ ba tìm được.

Ví dụ:

Input	Output	Input	Output	Input	Output
4	4	5	3	10	0
...D		..T..	AB....	
..C.		A....		..C....D..	
.B..		.FE.R		.E.....F.	
A...	X		...G..H...	
		S....		I.....J	
				K.....L	
				...M..N...	
				.O.....P.	
				..Q....R..	
			ST....	

Bài 5: APPLES (apples.*)

Bạn có một bức tranh hình chữ nhật có kích cỡ $m \times n$. Bức tranh này được chia thành $m \times n$ ô vuông bằng nhau, mỗi ô sẽ là một ô trống, hoặc là một quả táo, hoặc là một chướng ngại vật nào đó. Bạn muốn biết sau khi tất cả các quả táo đã rơi xuống hết mặt đất hoặc chướng ngại vật, bức tranh cuối cùng sẽ như thế nào. Để việc tìm bức tranh cuối cùng trở nên đơn giản, bạn quyết định sử dụng hai định luật sau, gọi là định luật Táo rơi I và Định luật táo rơi II:

+ Chướng ngại vật luôn đứng yên.

+ Nếu có một ô trống ở dưới 1 quả táo, quả táo sẽ di chuyển vào ô trống đó.

Hãy in ra bức tranh cuối cùng sau khi tất cả các quả táo đều đã rơi xong.

Dữ liệu vào:

+ Dòng đầu tiên chứa hai số nguyên dương m và n ($1 \leq m \leq 50000, 1 \leq n \leq 10$) cho biết số lượng dòng và cột của bức tranh.

+ m dòng tiếp theo, mỗi dòng gồm n kí tự mô tả bức tranh. Các ô trống được mô tả bằng kí tự '.', các ô chứa quả táo được mô tả bằng kí tự 'a', các ô chứa chướng ngại vật được mô tả bằng kí tự '#'.

Kết quả: Gồm m dòng, mỗi dòng chứa n ký tự mô tả bức tranh cuối cùng sau khi tất cả các quả táo đã rơi xong.

Ví dụ:

Input	Output
3 3	a..
aaa	#.a
#..	.a#
..#	

Input	Output
4 5
aaa.a	a....
aa.a.	aaaa.
a.a..	aaaaa
...a.	