

COCI – 10/2006 – Cuộc Thi I

01. PHÉP CHIA LẤY DƯ

Giới hạn bộ nhớ: 32MB

Cho 2 số nguyên A và B, khi chia A cho B ta lấy dư của phép chia đó. Ví dụ, các số 7, 14, 27 và 38 khi chia cho 3, lấy số dư sẽ được kết quả lần lượt là 1, 2, 0 và 2. Hãy viết 1 chương trình đọc 10 số (input) và xuất ra màn hình số các số dư khác nhau khi chia 10 số đó cho 42.

INPUT

Dữ liệu vào có 10 dòng. Mỗi dòng ghi một số nguyên không âm nhỏ hơn 100.

OUTPUT

Gồm 1 dòng duy nhất ghi số các số dư khác nhau khi chia 10 số đó cho 42.

Sample Input	Sample Output	Giải thích
1 2 3 4 5 6 7 8 9 10	10	Số các số dư là 10 bao gồm 1, 2, 3, 4, 5, 6, 7, 8, 9, 10.
42 84 252 420 840 126 42 84 420 126	1	Mọi số dư đều bằng 0.
39 40 41 42 43 44 82 83 84 85	6	Các số dư lần lượt là 39, 40, 41, 0, 1, 2, 40, 41, 0 và 1. Như vậy, có 6 giá trị khác nhau

02. NHÀ TOÁN HỌC HEMANN

Giới hạn bộ nhớ: 32MB

Vào thế kỉ thứ 19, nhà toán học người Đức Hemann Minkowski đã nghiên cứu thành công mặt phẳng phi Euclid, gọi là mặt phẳng taxicab. Trên mặt phẳng taxicab, khoảng cách giữa 2 điểm $T_1(x_1, y_1)$ và $T_2(x_2, y_2)$ được định nghĩa như sau:

$$D(T_1, T_2) = |x_1 - x_2| + |y_1 - y_2|$$

Và tất cả mọi định nghĩa khác đều đúng với mặt phẳng Euclid, kể cả định nghĩa một đường tròn:

Đường tròn là tập hợp của tất cả các điểm trong một mặt phẳng cách một điểm nào đó (tâm) một khoảng cách cố định (bán kính).

Chúng tôi quan tâm sự khác biệt giữa 2 đường tròn cùng bán kính R , một nằm trong mặt phẳng Euclidian (dạng bình thường), và cái kia nằm trong mặt phẳng taxicab. Gánh nặng về việc giải vấn đề khó khăn này đã được đặt lên bạn.

INPUT

Dòng duy nhất chứa độ dài bán kính R . $R \leq 100$. R là số nguyên.

OUTPUT

Ở dòng đầu tiên in ra diện tích của một đường tròn bán kính R trong mặt phẳng Euclid thường. Ở dòng tiếp theo in ra diện tích đường tròn với bán kính R trong mặt phẳng taxicab.

Chú ý: Dữ liệu ra chênh lệch ± 0.0001 so với đáp án chính thức vẫn được chấp nhận.

Sample Input	Sample Output
1	3.141593 2.000000
21	1385.442360 882.000000
42	5541.769441 3528.000000

03. OKVIRI

Giới hạn bộ nhớ: 32MB

“Khung Peter Pan” là cách trang trí các văn bản, với mỗi kí tự được đặt trong khung là một hình thoi, và khung của các kí tự cạnh nhau có thể đè lên nhau. Dưới đây minh họa khung Peter Pan bao quanh kí tự X:

```
..#..
.#.#.
#.X.#
.#.#.
..#..
```

Dù sao, với chỉ một khung hình như vậy trông sẽ có phần nhàm. Như vậy, cứ kí tự thứ 3, ta sẽ dùng khung hình Wendy. Khung hình Wendy có dạng như sau:

```
..*..
.*.*.
*.X.*
.*.*.
..*..
```

Khi khung hình Wendy xen kẽ cùng với khung hình Peter Pan thì khung hình của Wendy (đẹp hơn rất nhiều) sẽ được đặt lên trên.

INPUT

Gồm 1 dòng duy nhất chứaa nhiều nhất 15 kí tự chữ in hoa trong bảng chữ cái tiếng Anh.

OUTPUT

Với mỗi chữ cái trong từ được nhập vào, ta sẽ trang trí khung hình Peter Pan và Wendy trên 5 dòng.

Sample Input	Sample Output
A	..#.. .#.#. #.A.# .#.#. ..#..
DOG	..#...#...*... .#.#.#.#.*.*. #.D.#.O.*.G.* .#.#.#.#.*.*. ..#...#...*...
ABCD	..#...#...*...#... .#.#.#.#.*.*.#.#. #.A.#.B.*.C.*.D.# .#.#.#.#.*.*.#.#. ..#...#...*...#...

04. TRỞ VỀ

Giới hạn bộ nhớ: 32MB

Hoàng đế độc ác Cactus đang có trong tay chiếc chìa khóa ma thuật và làm lụt khu rừng Thần Tiên. Người họa sĩ và 3 chú nhím giờ đây phải trở lại chiếc hang của hải ly càng nhanh càng tốt, nơi mà bọn họ tránh xa khỏi dòng nước.

Bản đồ của khu rừng Thần Tiên chứa R hàng và C cột. Những cánh đồng trống được miêu tả bởi dấu '.', cánh đồng bị lụt là dấu '*' và những hòn đá là ký tự 'X'. Ngoài ra, hang trú ngụ của những chú hải ly là ký tự 'D', nơi người họa sĩ và 3 chú nhím đang đứng là nơi có ký tự 'S'.

Mỗi phút, người họa sĩ và 3 chú nhím có thể di chuyển 4 cánh đồng xung quanh (trên, dưới, trái, phải). Mỗi phút, trận lụt càng lan rộng ra cho nên tất cả các cánh đồng trống mà có ít nhất 1 phía là cánh đồng đã bị lụt thì nó cũng sẽ trở nên bị lụt giống như vậy. Cả dòng nước hay người họa sĩ cùng 3 chú nhím đều có thể đi qua các hòn đá. Hơn nữa, họ không thể vượt qua cánh đồng đã bị lụt, và nước không thể tràn vào hang của hải ly.

Viết một chương trình ghi ra thời gian ngắn nhất để cho những chú nhím và người họa sĩ đến hang của hải ly an toàn, nếu biết trước bản đồ của khu rừng Thần Tiên.

Chú ý: Người họa sĩ và 3 chú nhím không thể di chuyển đến một cánh đồng mà thời điểm đến cũng là thời điểm bị lụt trong cùng một phút.

INPUT

Dòng đầu tiên chứa 2 số nguyên R và C. ($R, C \leq 50$). R dòng tiếp theo chứa các ký tự như '.', '*', 'X', 'D' hoặc 'S'. Trên bản đồ chỉ có duy nhất một ký tự 'D' và một ký tự 'S'.

OUTPUT

Ghi ra thời gian ngắn nhất để người họa sĩ và 3 chú nhím đến được hang của hải ly một cách an toàn. Nếu không thể cho ra kết quả thì in ra từ "KAKTUS".

Sample Input	Sample Output	Giải thích
3 3 D.*S.	3	
3 3 D.*S	KAKTUS	Con đường tốt nhất họ có thể di chuyển là đi từng bước sang bên trái ở biên giới phía dưới, nhưng lại gặp cơn lụt 1 phút trước khi đến được cái hang.

3 6 D...*. .X.X..S.	6	
-----------------------------------	---	--

05. BOND

Giới hạn bộ nhớ: 32MB

Mọi người đều biết điệp viên bí ẩn 007 James Bond rất nổi tiếng. Thế nhưng một bí mật nhỏ là hầu hết nhiệm vụ anh chàng này không tự làm mà nhờ những người anh em họ có tên Jimmy Bonds. Ngài Bond (James Bond) đã rất mệt mỏi khi phải phân bổ nhiệm vụ cho Jimmy Bonds mỗi khi nhận được nhiệm vụ mới. Vì vậy, anh chàng đang muốn bạn giúp.

Mỗi tháng Bond (James Bond) nhận một danh sách các nhiệm vụ. Sử dụng trí thông minh cặn kẽ, tỉ mỉ từ những nhiệm vụ trước, với mọi nhiệm vụ và mọi anh chàng Jimmy Bond, James đều tính toán xác suất hoàn thành công việc cho mỗi một trong số họ. Chương trình của bạn sẽ xử lý lượng dữ liệu ấy và tìm ra cách sắp xếp để chất lượng công việc đạt được mức tốt nhất và mọi nhiệm vụ phải được hoàn thành đầy đủ.

Chú ý: Xác suất hoàn thành tất cả nhiệm vụ bằng với tích của tất cả các xác suất cho mỗi nhiệm vụ riêng lẻ được hoàn thành.

INPUT

Dòng đầu tiên chứa một số nguyên N, là số các anh em họ Jimmy Bonds và số nhiệm vụ ($1 \leq N \leq 20$). N dòng tiếp theo, mỗi dòng chứa N số nguyên từ 0 đến 100. Số nguyên thứ j trên dòng thứ i là xác suất mà Jimmy Bond i sẽ hoàn thành nhiệm vụ j, được coi là tỉ lệ phần trăm.

OUTPUT

Viết ra xác suất lớn nhất của các Jimmy Bond hoàn thành thành công tất cả nhiệm vụ dưới dạng tỉ lệ phần trăm. Lưu ý: Dữ liệu đầu ra chênh lệch ± 0.000001 so với đáp án chính thức vẫn được chấp nhận.

Sample Input	Sample Output	Giải thích
2 100 100 50 50	50.000000	
2 0 50 50 0	25.000000	
3 25 60 100 13 0 50 12 70 90	9.100000	Nếu Jimmy Bond số 1 nhận nhiệm vụ thứ 3, Jimmy Bond 2 nhận nhiệm vụ 1 và Jimmy Bond 3 nhận nhiệm vụ thứ 2 thì xác suất sẽ là: $1.0 * 0.13 * 0.7 = 0.091 = 9.1\%$. Mọi cách sắp xếp khác đều cho xác suất thành công nhỏ hơn.

06. SỬA LỖI

Giới hạn bộ nhớ: 32MB

Trong quá trình sửa lỗi chương trình, Mirko đã đề ý rằng một lỗi trong chương trình cho thể có liên quan đến sự tồn tại của “square killers” trong bộ nhớ chương trình. Bộ nhớ là ma trận R dòng và C cột, chỉ chứa các số 0 và 1. “square killer” là ma trận vuông nằm trong bộ nhớ, chứa nhiều hơn một kí tự, khi xoay 180 độ, ma trận đó trông vẫn như cũ. Ví dụ,

10101010	101...
11100101	...00.	111...
10100100.	101...
memory	killer	killer	killer

ma trận sau chứa 3 “square killer”:

Mirko đang nghĩ rằng có sự liên quan giữa kích cỡ của “square killer” lớn nhất và lỗi chương trình. Hãy giúp Mirko viết một chương trình, đọc bố trí của bộ nhớ, ghi ra kích thước “square killer” lớn nhất. Kích thước của mỗi “square killer” là số hàng hoặc số cột của ma trận vuông ấy. Trong ví dụ trên, kích thước của các “square killer” lần lượt là 2, 2 và 3.

INPUT

Dòng đầu tiên chứa 2 số nguyên R và C ($R, C \leq 300$). R dòng tiếp theo sẽ chứa C kí tự ‘0’ hoặc ‘1’, viết liền nhau.

OUTPUT

In ra kích cỡ của “square killer” lớn nhất. Nếu không có in ra -1.

Sample Input	Sample Output
3 6 101010 111001 101001	3
4 5 10010 01010 10101 01001	3
3 3 101 111 100	-1