

A – Số may mắn

Giới hạn thời gian: 1 giây

Chữ số 4 và 7 là các chữ số may mắn, trong khi các chữ số còn lại thì không được coi là may mắn. Một số nguyên là số may mắn nếu các chữ số của nó chỉ bao gồm chữ số 4 và 7. Nhiệm vụ của bạn là hãy viết một chương trình in ra số nguyên dương may mắn thứ K .

Input

Một dòng duy nhất là số nguyên K ($1 \leq K \leq 10^9$).

Output

In ra một dòng duy nhất là số nguyên dương may mắn thứ K .

Test ví dụ

Input	Output
1	4

Input	Output
2	7

Input	Output
3	4 4

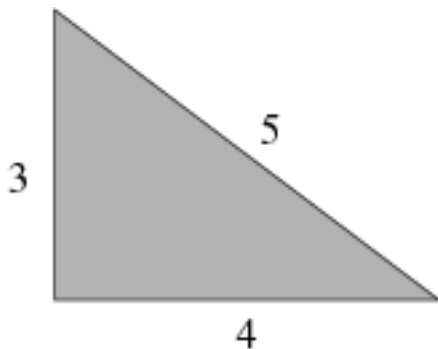
B – Định lý Pythagoras

Giới hạn thời gian: 1 giây

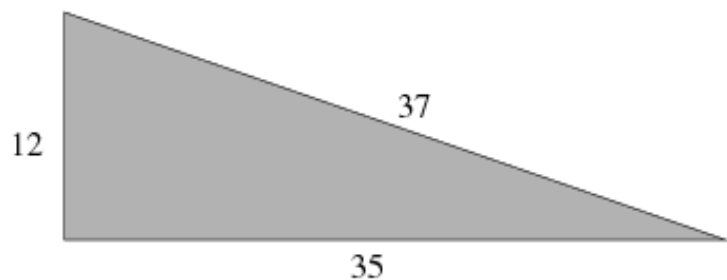
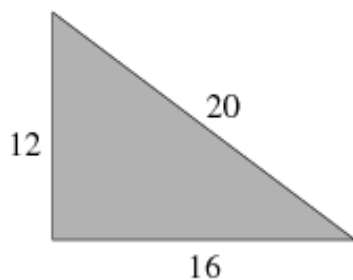
Định lý nổi tiếng mang tên nhà toán học người Hy Lạp Pythagoras phát biểu rằng trong một tam giác vuông thì bình phương độ dài cạnh huyền c bằng tổng bình phương độ dài hai cạnh góc vuông a và b .

$$a^2 + b^2 = c^2$$

Trên thực tế luôn tồn tại những tam giác vuông có độ dài của ba cạnh là các số nguyên, ví dụ như:



Một số ví dụ khác cho trường hợp $a = 12$ như:



Bạn được cho trước số nguyên a là độ dài của một cạnh góc vuông, nhiệm vụ của bạn là hãy viết một chương trình để đếm xem có bao nhiêu số nguyên b ($b > a$) là độ dài của cạnh góc vuông còn lại sao cho độ dài cạnh huyền c là một số nguyên.

Input

Dòng đầu tiên là một số nguyên T ($1 \leq T \leq 20$) là số test.

Mỗi test bao gồm một dòng duy nhất là số nguyên a ($2 \leq a < 2^{20}$) là độ dài của cạnh góc vuông nhỏ hơn.

Output

Với mỗi test in ra một dòng duy nhất là số lượng số nguyên b là độ dài của cạnh góc vuông lớn hơn tìm được sao cho độ dài cạnh huyền c là một số nguyên.

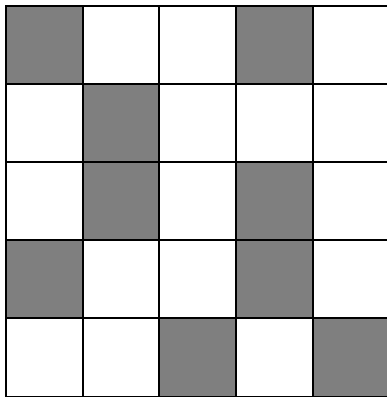
Test ví dụ

Input	Output
3	0
2	1
3	0
4	

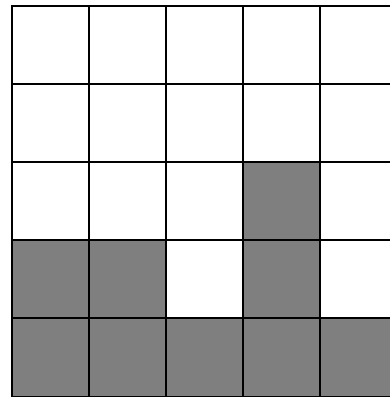
C – Hộp lập phương

Giới hạn thời gian: 1 giây

Bạn được cho một lưới ô vuông bao gồm m hàng và n cột. Một số ô của lưới ô vuông có chứa các hộp lập phương, và một ô chỉ có thể chứa được tối đa một hộp lập phương. Nếu tất cả các hộp lập phương của lưới ô vuông đều di chuyển theo phương thẳng đứng xuống dưới cho đến khi chúng không thể di chuyển được nữa, thì tất cả các hộp lập phương sẽ bị xếp chồng lên nhau ở đáy của lưới ô vuông. Hình 1 là một ví dụ về lưới ô vuông bao gồm 5 hàng và 5 cột, và có 9 ô có chứa các hộp lập phương. Sau khi di chuyển theo phương thẳng đứng xuống dưới, các hộp lập phương bị xếp chồng lên nhau ở đáy của lưới ô vuông như hình 2.



Hình 1. Lưới ô vuông với các hộp lập phương trước khi di chuyển



Hình 2. Lưới ô vuông với các hộp lập phương sau khi di chuyển

Khoảng cách di chuyển của một hộp lập phương được xác định bằng khoảng cách di chuyển của hộp lập phương đó từ vị trí ban đầu cho đến khi tất cả các hộp lập phương bị xếp chồng lên nhau ở đáy của lưới ô vuông. Trong ví dụ trên, khoảng cách di chuyển của hộp lập phương nằm ở góc bên trái phía trên cùng bằng 3, và tổng khoảng cách di chuyển của tất cả các hộp lập phương bằng 12.

Nhiệm vụ của bạn là hãy viết một chương trình để tính tổng khoảng cách di chuyển của tất cả các hộp lập phương của một lưới ô vuông cho trước.

Input

Dòng đầu tiên là một số nguyên T ($1 \leq T \leq 100$) là số test.

Với mỗi test, dòng đầu tiên bao gồm hai số nguyên m và n ($1 \leq m, n \leq 100$), trong đó m là số hàng và n là số cột của lưới ô vuông. Tiếp theo là m dòng, mỗi dòng bao gồm n số nguyên biểu diễn cho mỗi hàng của lưới ô vuông từ hàng trên cùng đến hàng dưới cùng. Nếu một ô có chứa hộp lập phương, ô đó có giá trị bằng 1. Ngược lại nếu một ô không chứa hộp lập phương, ô đó có giá trị bằng 0.

Output

Với mỗi test, in ra một dòng là tổng khoảng cách di chuyển của tất cả các hộp lập phương.

Test ví dụ

Input	Output
3	12
5 5	3
1 0 0 1 0	7
0 1 0 0 0	
0 1 0 1 0	
1 0 0 1 0	
0 0 1 0 1	
2 5	
1 0 1 0 1	
0 1 0 1 0	
4 4	
1 1 0 0	
1 0 1 0	
1 0 0 1	
0 1 0 1	

D – Bầu cử

Giới hạn thời gian: 1 giây

Cuộc bầu cử cho vị trí trưởng thôn sắp đến rất gần, Bờm đang có kế hoạch làm một chuyến đi xung quanh các xóm để vận động tranh cử, và sẽ có các bài diễn thuyết tại xóm Chài và xóm Lá. Vì lý do an ninh, hội đồng bầu cử muốn liên tục kiểm soát tất cả các xóm mà Bờm sẽ đến thăm (kể cả xóm Chài và xóm Lưới).

Dĩ nhiên, vì nguồn kinh phí hạn hẹp nên Bờm sẽ di chuyển đến các xóm bằng xe bò. Thêm nữa, hội đồng bầu cử sẽ sắp xếp chuyến đi của Bờm từ xóm Chài đến xóm Lá và quay trở về lại xóm Chài sao cho số lượng thành phố cần phải kiểm soát là ít nhất.

Với bài toán này, giả sử thôn có tất cả n xóm, được biểu diễn bởi các số từ 1 đến n , và m đường đi một chiều giữa các xóm với nhau, mỗi đường đi một chiều sẽ nối hai xóm khác nhau. Xóm Chài là xóm số 1 và xóm Lưới là xóm số 2.

Nhiệm vụ của bạn là hãy viết một chương trình xác định số lượng các thành phố cần phải được kiểm soát là ít nhất sao cho tồn tại một đường đi từ xóm Chài đến xóm Lưới và quay trở về lại xóm Chài, và chỉ đi qua những thành phố được kiểm soát.

Input

Dòng đầu tiên bao gồm hai số nguyên n và m ($2 \leq n \leq 100, 2 \leq m \leq 200$) lần lượt là số lượng các xóm và số lượng các đường nối giữa các xóm.

Tiếp theo là m dòng, mỗi dòng bao gồm hai số nguyên a và b ($1 \leq a, b \leq n, a \neq b$) là đường đi có hướng từ xóm a đến xóm b . Không tồn tại hai đường đi nối cùng hai xóm theo cùng một hướng.

Output

In ra một dòng duy nhất là số lượng ít nhất các thành phố cần phải được kiểm soát. Dữ liệu đầu vào đảm bảo bài toán luôn có kết quả.

Test ví dụ

Input	Output
12 10 1 2 2 3 3 4 4 5 5 6 6 7 7 8 8 9 9 10 10 1	10

E – Đa giác đều

Giới hạn thời gian: 1 giây

Đa giác đều là một đa giác mà tất cả các cạnh của nó có độ dài bằng nhau, và tất cả các góc trong của đa giác nhỏ hơn hoặc bằng 180° . Ví dụ hình vuông là một đa giác đều. Bạn được cho trước tọa độ của ba điểm là ba đỉnh của một đa giác đều, nhiệm vụ của bạn là hãy viết một chương trình xác định số lượng đỉnh nhỏ nhất của đa giác đều đó.

Input

Dòng đầu tiên là một số nguyên T ($1 \leq T \leq 1500$) là số test.

Mỗi test bao gồm ba dòng, mỗi dòng bao gồm hai số thực x_i và y_i ($-10^4 \leq x_i, y_i \leq 10^4$), trong đó (x_i, y_i) là tọa độ của một đỉnh của đa giác đều. Tọa độ của các điểm được cho với độ chính xác là 10^{-6} . Biết rằng trong tất cả các test, khoảng cách Euclidean giữa hai điểm bất kỳ luôn lớn hơn hoặc bằng 1, và đa giác đều có nhiều nhất là 1000 đỉnh.

Output

Với mỗi test, in ra một dòng là số lượng đỉnh nhỏ nhất của đa giác đều.

Test ví dụ

Input	Output
2	5
4806.0687720 -6918.0168276	3
793.5131567 8653.7751792	
-7593.0266050 3321.3065241	
-5685.8777725 6182.9632793	
-3505.8591555 -8289.5976147	
7937.7369280 834.6343354	

F – Con ốc sên

Giới hạn thời gian: 1 giây

Có một con ốc sên muốn bò lên đỉnh của một cái cây cao V mét tính từ mặt đất. Trong một ngày nó có thể bò được A mét lên trên, tuy nhiên mỗi đêm khi ngủ, nó lại bị tụt xuống B mét. Nhiệm vụ của bạn là hãy viết chương trình xác định số ngày con ốc sên cần để bò lên đến đỉnh cây.

Input

Một dòng duy nhất bao gồm ba số nguyên A, B và V ($1 \leq B < A \leq V \leq 10^9$).

Output

In ra một dòng duy nhất là số ngày con ốc sên cần để bò lên đến đỉnh cây.

Test ví dụ

Input	Output
2 1 5	4

Input	Output
5 1 6	2

G – Bộ sưu tập phim

Giới hạn thời gian: 1 giây

Bờm có một bộ sưu tập phim rất lớn. Bờm quản lý bộ sưu tập của mình bằng cách xếp các đĩa phim chồng lên nhau tạo thành một chồng đĩa lớn. Mỗi lần Bờm muốn xem một bộ phim nào đấy, cậu ta xác định vị trí của đĩa phim trong chồng đĩa và lấy nó ra thật cẩn thận để đảm bảo rằng chồng đĩa không bị đổ xuống. Sau khi Bờm xem xong phim, cậu ta xếp trả lại đĩa phim vừa xem ở trên cùng của chồng đĩa.

Vì chồng đĩa phim rất lớn nên Bờm muốn kiểm soát và theo dõi vị trí của từng đĩa phim. Bờm muốn biết với mỗi đĩa phim thì có bao nhiêu đĩa phim được xếp chồng lên phía trên đĩa phim đó, và từ thông tin này Bờm có thể biết được vị trí của đĩa phim trong chồng đĩa. Mỗi đĩa phim được xác định bởi một số hiệu được in lên hộp đựng đĩa.

Nhiệm vụ của bạn là hãy viết một chương trình để giúp Bờm kiểm soát và theo dõi vị trí của từng đĩa phim. Cụ thể hơn, khi Bờm lấy một đĩa phim ra khỏi chồng đĩa, chương trình của bạn phải in ra số lượng đĩa phim được xếp chồng lên phía trên đĩa phim đó trước khi nó bị lấy ra.

Input

Dòng đầu tiên là một số nguyên T ($1 \leq T \leq 100$) là số test.

Với mỗi test, dòng đầu tiên là hai số nguyên n và m ($1 \leq n, m \leq 100000$), trong đó n là số lượng đĩa phim và m là số lần lấy đĩa phim ra khỏi chồng đĩa để xem của Bờm. Dòng thứ hai bao gồm m số nguyên a_1, a_2, \dots, a_m ($1 \leq a_i \leq n$) là số hiệu của các đĩa phim mà Bờm muốn xem.

Để đơn giản hóa bài toán, giả sử chồng đĩa ban đầu chứa các đĩa phim với số hiệu lần lượt là $1, 2, \dots, n$ theo thứ tự tăng dần, với đĩa phim có số hiệu 1 nằm ở trên cùng.

Output

Với mỗi test, in ra một dòng bao gồm m số nguyên, số nguyên thứ i cho biết số lượng đĩa phim được xếp chồng lên phía trên đĩa phim có số hiệu a_i ngay trước khi nó bị lấy ra khỏi chồng đĩa. Chú ý sau mỗi lần đĩa phim có số hiệu a_i bị lấy ra khỏi chồng đĩa, sau đó nó được xếp trả lại ở trên cùng.

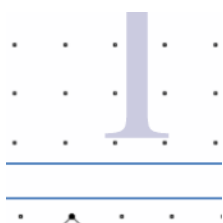
Test ví dụ

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

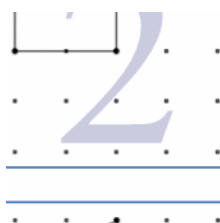
H – Hình vuông tọa độ nguyên

Giới hạn thời gian: 5 giây

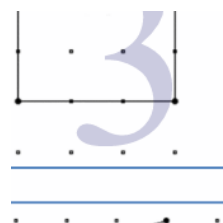
Hình vuông tọa độ nguyên là một hình vuông có hoành độ và tung độ của các đỉnh trong hệ tọa độ Cartesian là các số nguyên. Ví dụ (3, 4) là một điểm có tọa độ nguyên, (2, 2.5) không phải là một điểm có tọa độ nguyên. Trong một lưới có kích thước $m \times n$ có thể có rất nhiều các hình vuông tọa độ nguyên. Hình 1, 2, 3, 4, 5 và 6 là một số ví dụ về các hình vuông tọa độ nguyên trong lưới có kích thước 4×4 . Đếm số lượng hình vuông tọa độ nguyên có các cạnh song song với các trục tọa độ (hình 1, 2 và 3) là một việc tương đối dễ dàng, tuy nhiên vẫn tồn tại các hình vuông tọa độ nguyên mà các cạnh của chúng không song song với các trục tọa độ do đó việc đếm chúng sẽ khó khăn hơn một chút. Một số hình vuông có thể có diện tích chẵn (hình 2, 4 và 6) và một số khác có thể có diện tích lẻ (hình 1, 3 và 5).



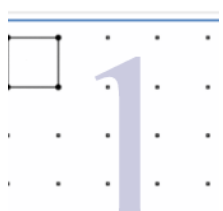
Hình 1. Hình vuông tọa độ nguyên có diện tích bằng 1



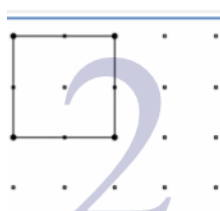
Hình 2. Hình vuông tọa độ nguyên có diện tích bằng 4



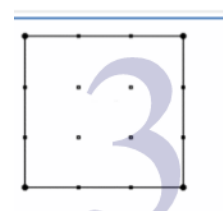
Hình 3. Hình vuông tọa độ nguyên có diện tích bằng 9



Hình 4. Hình vuông tọa độ nguyên có diện tích bằng 2



Hình 5. Hình vuông tọa độ nguyên có diện tích bằng 5



Hình 6. Hình vuông tọa độ nguyên có diện tích bằng 10

Bạn được cho trước một lưới có kích thước $m \times n$, nhiệm vụ của bạn là hãy viết một chương trình để đếm số lượng hình vuông tọa độ nguyên khác nhau có diện tích lẻ có thể vẽ được trong lưới.

Input

Dòng đầu tiên là một số nguyên T ($1 \leq T \leq 50000$) là số test.

Mỗi test là một dòng duy nhất bao gồm hai số nguyên m và n ($1 \leq m, n \leq 100000$) là kích thước của lưới.

Output

Với mỗi test, in ra một dòng duy nhất là số lượng hình vuông tọa độ nguyên khác nhau có diện tích lẻ tìm được trong lưới, biết rằng số lượng này có độ lớn nằm trong phạm vi 64-bit.

Test ví dụ

Input	Output
2	2
1 2	12
3 3	

I – Đếm hình vuông

Giới hạn thời gian: 2 giây

Bạn được cho n đường thẳng khác nhau trong hệ trục tọa độ Oxy, nhiệm vụ của bạn là hãy đếm số lượng hình vuông có thể tạo bởi n đường thẳng này.

Input

Dòng đầu tiên là một số nguyên n ($1 \leq n \leq 2000$) là số đường thẳng.

Tiếp theo là n dòng, mỗi dòng bao gồm bốn số nguyên x_1, y_1, x_2 và y_2 ($|x_1|, |y_1|, |x_2|, |y_2| \leq 10000$) lần lượt là tọa độ của hai điểm khác nhau (x_1, y_1) và (x_2, y_2) mà đường thẳng thứ i đi qua.

Output

In ra một dòng duy nhất là số lượng hình vuông có thể tạo bởi n đường thẳng này.

Test ví dụ

Input	Output
10 0 0 1 0 0 1 1 1 0 2 2 2 0 0 0 4 1 -1 1 0 2 -2 2 2 1 1 2 2 1 1 0 2 3 1 2 2 1 3 0 2	6

J – Tính chẵn lẻ

Giới hạn thời gian: 1 giây

Bạn được cho một số nguyên n , nhiệm vụ của bạn là hãy viết một chương trình kiểm tra xem số nguyên đó là số chẵn hay số lẻ.

Input

Dòng đầu tiên là một số nguyên T ($1 \leq T \leq 20$) là số test.

Tiếp theo là T dòng, mỗi dòng là một số nguyên n ($-10 \leq n \leq 10$).

Output

Với mỗi test, nếu n là số chẵn thì in ra `n is even`, ngược lại nếu n là số lẻ thì in ra `n is odd`.

Test ví dụ

Input	Output
3	7 is odd
7	6 is even
6	-4 is even
-4	

K – Xếp hàng

Giới hạn thời gian: 1 giây

Có n người đang đứng xếp hàng để đi vào một buổi hòa nhạc. Vì cảm thấy chán nản do chờ đợi quá lâu, họ bắt đầu nhìn ra những người khác xung quanh mình.

Hai người A và B đứng xếp hàng có thể nhìn thấy nhau nếu như họ đứng ngay cạnh nhau hoặc ở giữa họ không có ai cao hơn người A hoặc người B.

Nhiệm vụ của bạn là hãy viết một chương trình xác định số cặp người có thể nhìn thấy nhau.

Input

Dòng đầu tiên là một số nguyên n ($1 \leq n \leq 500000$) là số người đang đứng xếp hàng.

Tiếp theo là n dòng, dòng thứ i là một số nguyên a_i ($1 \leq a_i < 2^{31}$) là chiều cao của người đứng thứ i trong hàng.

Output

In ra một dòng duy nhất là số cặp người có thể nhìn thấy nhau.

Test ví dụ

Input	Output
7 2 4 1 2 2 5 1	10