

Kỳ thi thử Duyên Hải lần 2

A. PARALLEL

256MB | 2s

Ban giám khảo đã chuẩn bị rất nhiều bài thú vị. Nhưng họ không thích chủ tịch ban giám khảo. Ông này cho rằng tất cả các bài đều quá khó. Và để cho thí sinh giải được ít nhất một bài, đề thi cần một bài dễ.

Chủ tịch hội đồng giám khảo gọi bài là dễ là nếu lời giải của nó sử dụng không nhiều hơn 1 lệnh rẽ nhánh và không nhiều hơn 2 vòng lặp hoặc sử dụng không nhiều hơn 2 lệnh rẽ nhánh và không nhiều hơn 1 vòng lặp. Hội thẩm đưa ra n bài toán khác nhau với lời giải tương ứng và nộp cho chủ tịch hội đồng giám khảo. Ông tính toán số lượng các lệnh rẽ nhánh và các vòng lặp trong lời giải của từng bài và bây giờ muốn biết bài nào là dễ.

Bạn có n bài, mỗi bài bạn biết số lượng các lệnh rẽ nhánh và số lượng các vòng lặp trong lời giải. Hãy xác định xem bài có dễ không.

INPUT

Dòng đầu tiên ghi số lượng bài n ($1 \leq n \leq 121$). Tiếp theo, mỗi dòng trong n dòng tiếp theo ghi 2 số nguyên i và f ($0 \leq i, f \leq 10$) - số lệnh rẽ nhánh và số vòng lặp được sử dụng trong lời giải của một bài.

OUTPUT

Đối với mỗi bài, in ra trên một dòng "Yes" nếu bài dễ, và "No" nếu ngược lại.

Sample Input	Sample Output
2	Yes
2 1	No
2 2	

B. KHÓA THEN CHỐT

256MB | 2s

Để mã hóa một khóa s không rỗng, đầu tiên điệp viên chọn một chuỗi t sao cho chuỗi s là một chuỗi tiền tố của t , và chuỗi nghịch đảo của s là hậu tố của chuỗi t . Trong trường hợp này, chuỗi t có thể ghi các ký hiệu không liên quan đến chuỗi s . Sau đó, một số ngẫu nhiên m (có thể $m = 0$) ký tự ngẫu nhiên được đưa vào bên trái của t , mà m ký tự ngẫu nhiên được thêm vào bên phải t . Bây giờ, chuỗi t là mã hóa của s .

Rõ ràng với t , có thể có nhiều s . Do đó, người ta quyết định rằng khóa s phải là chuỗi dài nhất trong tất cả các tùy chọn có thể, và trong trường hợp có nhiều tùy chọn dài nhất, chọn số lượng các ký tự ngẫu nhiên thêm vào bên trái và vào bên phải là tối thiểu, nghĩa là, chuỗi t có chiều dài tối đa. Bạn cần phải thực hiện thuật toán phục hồi khóa s từ phiên bản được mã hóa t .

INPUT

Dòng đầu tiên ghi số nguyên n - số khóa mà bạn cần phải giải mã. Mỗi dòng trong n dòng sau là một chuỗi mã hóa t , chỉ gồm chữ cái thường trong bảng chữ cái Latinh. Tổng chiều dài của tất cả các khóa không vượt quá 100.000 ký tự. Đảm bảo rằng mỗi phiên bản mật mã có ít nhất một khóa không rỗng.

OUTPUT

Với mỗi bản mã t , in ra trên một dòng khóa s tương ứng.

Sample Input	Sample Output
3	bab
ababc	ababa
ababa	xba
cxbaydzabxe	

C. TÓM TẮT

256MB | 2s

Trái ngược với câu nói nổi tiếng "diêm không phải là đồ chơi của trẻ con", bé An vẫn rất thích chơi diêm. Nhưng bé không nghịch lửa mà dùng diêm để xếp hình.

Gần đây, mẹ cho bé một số hộp diêm, mỗi bộ gồm sáu que diêm. Bé bắt đầu tự hỏi: liệu có thể dùng keo, dính các que diêm để tạo thành một hình tứ diện được không? Không được bẻ que diêm, các que diêm không được thò ra ngoài.

INPUT

Dòng đầu tiên ghi số hộp diêm n ($1 \leq n \leq 1000$). Mỗi dòng trong n dòng tiếp theo ghi 6 số nguyên trong khoảng 1-1000 là độ dài một que diêm trong hộp

OUTPUT

Với mỗi hộp diêm, in ra trên một dòng Yes nếu có thể tạo thành hình tứ diện, in ra No nếu không thể.

Sample Input	Sample Output
4	Yes
1 1 1 1 1 1	No
1 2 3 1 2 3	Yes
1 2 1 2 1 2	Yes
1 2 2 1 2 2	

D. CPU

DMA mới phát minh ra một loại vi xử lý hai nhân kiểu mới có thể thực hiện 26 chỉ thị khác nhau, kí hiệu bằng 26 chữ cái in hoa trong bảng chữ cái tiếng Anh. Trong mỗi xung nhịp, một nhân của vi xử lý có thể xử lý đúng một chỉ thị. Một chương trình sau khi được biên dịch sẽ gồm một chuỗi các chỉ thị cần được thực thi tuần tự. Vì là vi xử lý hai nhân, nên người ta có thể chạy hai chương trình cùng một lúc. Tuy nhiên, loại vi xử lý này có một nhược điểm là không thể thực hiện hai chỉ thị khác nhau trên hai nhân trong cùng một xung nhịp. Dĩ nhiên, vi xử lý sẽ tìm cách tối ưu để thực thi được cả hai chương trình trong số lượng xung nhịp ít nhất.

Ví dụ với 2 chương trình ABB và ABC, vi xử lý sẽ thực hiện đồng thời hai chỉ thị A trong xung nhịp thứ nhất, hai chỉ thị B trong xung nhịp thứ hai, chỉ thị B của chương trình thứ nhất trong xung nhịp thứ ba và chỉ thị C của chương trình thứ hai trong xung nhịp thứ tư. Như vậy, vi xử lý cần 4 xung nhịp để thực thi xong cả hai chương trình. Và đây là phương án tối ưu nhất.

An đã mua về n vi xử lý loại vừa nêu và hiện có $2n$ chương trình cần thực thi. Hãy giúp cậu ấy phân $2n$ chương trình này cho n máy chạy song song, mỗi máy chạy hai chương trình sao cho thời gian thực thi là nhỏ nhất có thể.

INPUT

Dòng đầu tiên chứa số nguyên n ($1 \leq n \leq 10$) là số lượng vi xử lý. $2n$ dòng sau, mỗi dòng chứa từ 1 đến 100 kí tự in hoa trong bảng chữ cái tiếng Anh mô tả chuỗi chỉ thị của một chương trình.

OUTPUT

Ghi ra thời gian thực thi tối thiểu trên một dòng duy nhất.

Sample Input	Sample Output
2 ABB BAB CAB ABC	4

E. NGƯỜI SOẠN THẢO VĂN BẢN

256MB | 2s

Bé An quyết định tự viết IDE cho mình. Và một trong những chức năng rất quan trọng là các cặp dấu ngoặc.

Ở giai đoạn đầu, IDE chỉ nên hỗ trợ một thao tác - thay đổi ký tự ở vị trí i . Tuy nhiên, IDE không cho phép văn bản dài hơn n ký tự. Mỗi lần nếu có ký tự mới là một dấu ngoặc mở hoặc dấu ngoặc đóng, IDE phải làm nổi bật dấu ngoặc tương ứng đóng hoặc mở tương ứng.

Chúng ta hãy định nghĩa khái niệm dấu ngoặc mở. Giả sử dấu ngoặc mở ở vị trí i trong văn bản. Thì dấu ngoặc đóng ứng với nó là dấu ngoặc đóng tại vị trí j , thỏa mãn:

- $i < j$;
- nếu lấy đoạn văn bản từ vị trí i đến vị trí j và loại bỏ tất cả các ký hiệu không phải là dấu ngoặc, chúng ta sẽ thu được một dãy ngoặc đúng
- j có giá trị bé nhất có thể.

Định nghĩa tương tự cho dấu ngoặc đóng.

Hãy giúp An xác định các ngoặc tương ứng của nhau.

INPUT

Dòng đầu tiên ghi số nguyên n ($1 \leq n \leq 100000$) - độ dài tối đa của văn bản, và m ($1 \leq m \leq 100000$) - số lần thao tác sửa đổi ký hiệu.

Mỗi dòng trong m dòng tiếp theo mô tả một thao tác sửa đổi có dạng $i c$: thay đổi ký tự ở vị trí i bằng ký tự c ($1 \leq i \leq n$, c là chữ thường trong bảng chữ cái Latinh hoặc dấu ngoặc đơn). Ban đầu, văn bản có n chữ cái Latin "a".

OUTPUT

Đối với mỗi thao tác thay đổi ký tự bằng dấu ngoặc đơn, in ra trên một dòng vị trí của dấu ngoặc đơn tương ứng với nó. Nếu không tồn tại, in -1.

Sample Input	Sample Output
3 4	-1
1 (1
3)	1

2)	-1
3)	