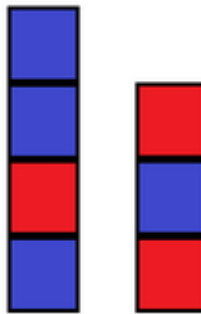


## Problem D. Torres bellísimas

**Time limit** 2000 ms

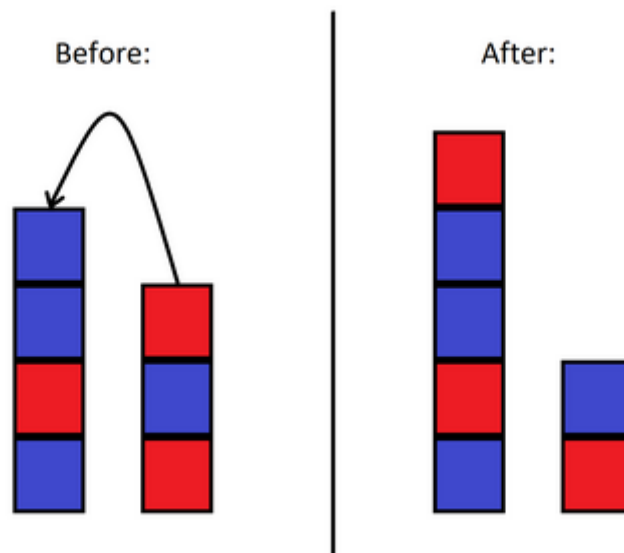
**Mem limit** 262144 kB

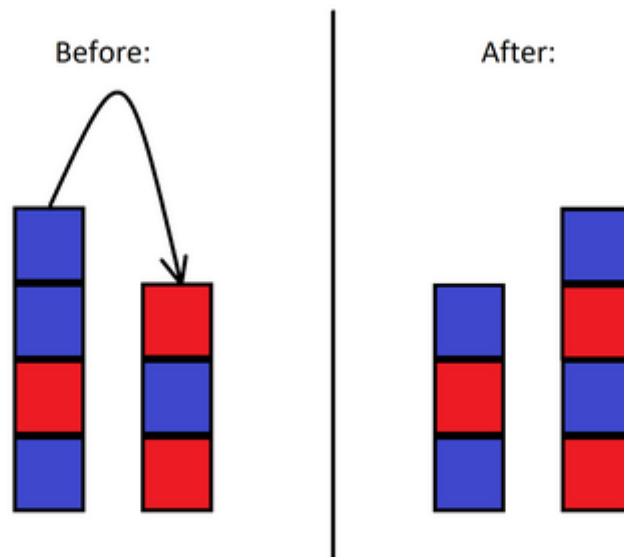
Hay dos torres con bloques de dos colores: rojo y azul. Ambas torres son representadas por un string de caracteres  $R$  y  $B$  (por red y blue) que denotan el orden de los bloques de abajo para arriba.



Estas dos torres son representadas por los strings  $BRBB$  y  $RBR$ .

Puedes hacer la siguiente operación cualquier cantidad de veces: elije una torre con al menos dos bloques y mueve el bloque superior al tope de la otra torre.





Las dos torres son *bellísimas* si ningún par de bloques adyacentes tiene el mismo color, o sea, ningún bloque rojo está sobre otro bloque rojo ni ningún bloque azul esta sobre otro azul.

Tienes que revisar si es posible hacer que las torres sean *bellísimas* después de cualquier cantidad de operaciones (posiblemente ninguna).

### Input

La primera línea tiene un entero  $t$  ( $1 \leq t \leq 1000$ ) — la cantidad de casos de prueba.

Cada caso de prueba tiene tres líneas:

- la primera línea tiene dos enteros  $n$  y  $m$  ( $1 \leq n, m \leq 20$ ) — la cantidad de bloques en la primera y segunda torre respectivamente;
- la segunda línea tiene a  $s$  — un string de exactamente  $n$  caracteres B y/o R, que representa a la primera torre;
- la tercera línea tiene a  $t$  — un string de exactamente  $m$  caracteresB y/o R, que representa a la segunda torre.

### Output

Para cada caso de prueba, imprime YES si es posible que las torres queden *bellísimas* después de alguna cantidad de operaciones, imprime NO en el caso contrario.

### Sample 1

Input	Output
4	YES
4 3	YES
BRBB	YES
RBR	NO
4 7	
BRBR	
RRBRBRB	
3 4	
RBR	
BRBR	
5 4	
BRBRR	
BRBR	

**Note**

En el primer caso, puedes mover el tope de la primera torre a la segunda torre.

En el segunda caso, puedes mover el tope de la segunda torre a la primera torre 6 veces seguidas.

En el tercer caso, las torres ya son *bellísimas*.