G - Sequential Consistency

Context

When executing a parallel application, the programmer expects that every instruction is executed in the order they appear in the program. If this happens, the result of the application is sequentially consistent. A parallel application can give different sequentially consistent results, depending on the interleaving of instructions in the different threads.

The Problem

To know all the possible results of a program assuming sequential consistency, all possible interleavings of the instructions in different threads have to be tested. For example, given the following program, where one thread executes the first column and the other thread executes the second column:

```
a = 1;       b = 1;
print b;     print a;
```

The possible interleavings are the following:

a = 1;
print b;
b = 1;
print a;
	a = 1;
	b = 1;
	print b;
	print a;
		b = 1;
		a = 1;
		print b;
		print a;
			a = 1;
			b = 1;
			print a;
			print b;
				b = 1;
				a = 1;
				print a;
				print b;
					b = 1;
					print a;
					a = 1;
					print b;

Assuming that all variables are initialized to 0, the possible results are:

01	11	11	11	11	01

Your task is to write a program that given a printed sequence, tells you if the program was executed following sequential consistency, that is, respecting the order of the instructions in the thread.

The Input

The program input begins with a number indicating the number of programs. The programs follow in the next lines. A program has always two threads. Each thread has a number of instructions less than 10, and each instruction goes in a different line. Each instruction can be of the type "variable = value" or of the type "print variable". Variables are represented by single character (from 'a' to 'z') and values are a single digit (from '0' to '9'). The code of each thread finishes with the character hash ('#'). After the code, each line represents a possible result. The list of results ends with the character hash ('#').

The Output

Your program must print for each possible result, if the program can print it (YES) or not (NO), assuming sequential consistency. An empty line separates outputs from different programs.

Sample Input

```
2
a = 1
print b
#
b = 1
print a
#
00
01
11
10
111
#
print a
#
print b
#
11
00
#
```

Sample Output

```
NO
YES
YES
NO
NO

NO
YES
```