Ball of Whacks
Problem ID: ballofwhacks

Danny has a new toy called Ball of Whacks. It's a rhombic triacontahedron (which we all know is a thirty sided polyhedron) made of small pyramid-shaped pieces which look like the following:
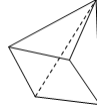

Figure 1: Game piece.

When the pieces are all put together, they form the object shown in Figure 2a. Figures 2b-2f show how the rhombic triacontahedron is built: at the bottom is a petal-shaped section of five pyramids (2b — note that only the base of each pyramid piece is shown); on top of that is a ring of five more pieces (2c); on top of that is a larger ring of ten pieces (2d), following by a five-piece ring (2e) and another petal-shaped section (2f). Figure 3 shows how the pieces are related to one another.

Danny has several friends who also own Balls of Whacks, and when they get together...well, balls get whacked. The result are several large sections lying on the floor. When it's time to go home, Danny needs to know which sections fit together to make one complete Ball of Whacks. He would like you to help him.
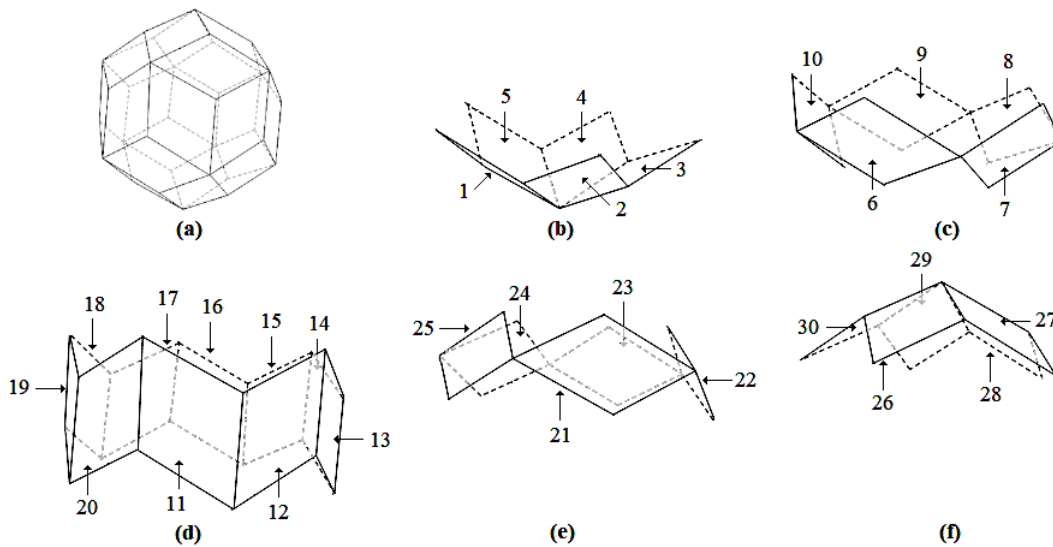

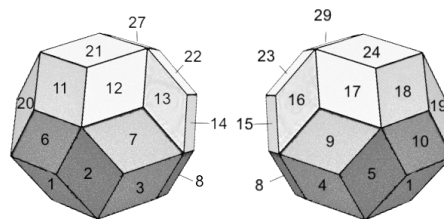Figure 2: Construction of a Ball of Whacks.

Figure 3: Numbering scheme of an assembled Ball of Whacks

Input

The input file consists of three lines each containing a description of a Ball of Whacks section. Each description starts with an integer m indicating the number of pieces in the section, followed by a set of numbers between 1 and 30 indicating the relative alignment of the pyramid pieces in the section, using the numbered locations in diagrams in Figure 2b-2f. Each section description starts with a piece in position 1, though that piece — along with all other pieces in the section — may need to be moved in order for all three sections to fit together. The pieces in any section will be connected, i.e. you can get from any one piece in a section to any other by crossing over pieces which share an internal face.

Output

Output either `Yes` if the three sections can be fit together to get one Ball of Whacks, or `No` if they cannot be fit together.

Sample Input 1	Sample Output 1

`25 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25`
`3 1 2 3`
`2 1 2`
	`Yes`

Sample Input 2	Sample Output 2

`24 1 4 5 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 24 25 26 27 30`
`5 1 2 3 4 5`
`1 1`
	`No`