

ASCII Dancer

Time limit: 2500 ms
Memory limit: 256 MB

In this challenge, we will create a dancing ASCII man. You will be given a series of commands that indicate either the moves the man makes or the things the man says.

If an input line starts with the command "say ", you will output the rest of the line.

The other commands cause the current position of the man to be output. The man starts facing forward in the following position:

```
  | o
2  /|\
3  / \
4
```

There are four commands cause the man to move his left hand. If he is in the starting position, facing forward, they will result in the following output:

"left hand to head"

```
  | o)
2  /|
3  / \
4
```

"left hand to hip"

```
  | o
2  /|>
3  / \
4
```

"left hand to start"

```
  | o
2  /|\
3  / \
4
```

Note that you would output the exact same output if the man was facing backwards and the commands referred to the "right hand" instead of the "left hand". You would output the mirror image if the man was facing backwards and given the same commands, or if the man is facing forwards and the commands referred to the "right hand" instead of the "left hand".

The command "turn" causes the ASCII man to change the way he is facing, for example, facing backwards if he was facing forwards, or facing forwards if he was facing backwards.

There are two commands cause the man to move his right leg. If he is in the starting position, facing forward, they will result in the following output:

"right leg in"

```
  | o
2  /|\
3  < \
4
```

"right leg out"

```
  | o
2  /|\
3  / \
4
```

If a previous limb has moved, it stays in the same position until it moves again. That allows for sequences of moves like the one below. Here we are starting from the starting position, facing forwards:

"right hand to hip"

```
  | o
2  <|\
3  / \
4
```

"left hand to head"

```
  | o)
2  <|
3  / \
4
```

"left leg in"

```
2 | o)
3 <|
4 / >
```

Standard input

The input begins with an integer T giving the number of test cases.

Each test case begins with an integer d that gives the number of commands in the dance.

The next d lines give a command in from the following list:

- "left hand to head"
- "left hand to hip"
- "left hand to start"
- "left leg in"
- "left leg out"
- "right hand to head"
- "right hand to hip"
- "right hand to start"
- "right leg in"
- "right leg out"
- "turn"
- "say [words]", where [words] is a string of made up of letters, numbers, punctuation, and spaces.

Standard output

Output consists of either three lines, with three characters each, in response to a movement command, or one line in response to a "say" command.

Constraints and notes

- $1 \leq T \leq 10$
- $3 \leq d \leq 400$

The number of characters to be output in any "say" command will be between 1 and 80, inclusive.

It is possible that a move command will not change the position of the ASCII Man, for example if the same command is given twice in a row. Even if the position does not change as a result of a movement command, you still should produce the output showing the current position.

The input will never ask to have both the left leg in and the right leg in.

Input	Output	Explanation
<pre>1 19 say Hokey Pokey say You put your right foot in right leg in say You put your right foot out right leg out say You put your right foot in right leg in say Shake it all about left hand to head right hand to hip left hand to hip right hand to head say You do the hokey pokey and you turn yourself around turn turn say That's what it's all about! right leg out</pre>	<pre>Hokey Pokey You put your right foot in o / \ < \ You put your right foot out o / \ / \ You put your right foot in o / \ < \ Shake it all about o) / < \ o) < </pre>	The sample input contains a single testcase with a dance that contains 19 commands.

right hand to start
left hand to start

```
< \  
  o  
<|>  
< \  
(o  
  |>  
< \  
You do the hokey pokey and you turn yourself around  
  o)  
<|  
/ >  
(o  
  |>  
< \  
That's what it's all about!  
(o  
  |>  
/ \  
  o  
/|>  
/ \  
  o  
/|\   
/ \
```