

Problem F Smart Sushi

Time limit: 3 seconds

Memory limit: 1024 megabytes

Problem Description

Professor Shih-Tsung loves eating sushi (maybe). One day, he went to a restaurant called Smart Sushi. This restaurant serves sushi on plates of different colors, and each color represents a different price.

Just as the professor was about to finish his meal and pay the bill, the clerk excitedly announced a special event called “Sushi Hanoi.”

In this event, there are three positions available: A, B, and C. In the beginning, there are n plates of different colors stacked on position A, the plate whose color represents the cheapest price is placed on top, while the plate whose color represents the most expensive price is at the bottom. The goal is to move all plates from A to C, following the similar rules as in the Tower of Hanoi:

- Only one plate can be moved at a time.
- The plates whose color represents the cheaper price should always be placed on the top of the plates whose color represents the more expensive price.

The clerk tells Professor Shih-Tsung that if he can complete the challenge in fewer moves than the clerk, he will get 50% off his meal!

However, the professor is an expert in such puzzles – so to help the clerk winning the game, please write a program that prints the exact sequence of moves in minimum operation times to transfer all plates from A to C following the rules.

Input Format

Your program is to read from standard input. The input consists of several test cases. Each test case contains only one integer number n . The input ends with 0.

Output Format

Your program is to write to standard output. For each test case, please output the moving step in ‘step – source to target’ format, for example, if you move the top plate from A to B at step 2, then print ‘2 – A to B’. Note that there is a space between step and dash, dash and source, source and ‘to’, ‘to’ and target. Different test cases are separated by a blank line. Please see the sample output.

Technical Specification

- $1 \leq n \leq 20$

Sample Input 1

```
1  
3  
0
```

Sample Output 1

```
1 - A to C  
1 - A to C  
2 - A to B  
3 - C to B  
4 - A to C  
5 - B to A  
6 - B to C  
7 - A to C
```