

Draw Mountains

You are a member of a team that is working in a new wonderful graphics program. Your task is to write a module for drawing skylines of mountains. However, for now the program is in beta version, so each skyline is discretized. This means that your module receives as input a sequence of integers representing heights. Each pair of consecutive heights in the sequence may differ by at most 1 and produces a column of output, each column containing a single character. The particular character to write and its position in the column depend on the pair of heights. When both heights are equal an underscore (“_”) must be printed. If the heights within the pair increase or decrease, you must write a slash (“/”) or a backslash (“\”), respectively. The position of the character in the column is such that the skyline visually respect the given heights. For instance, the sequence of heights (1, 2, 3, 2, 3, 3, 2, 1, 0) must produce the skyline shown in the following figure.

```
  _  
 /  \  
/    \  
 \   \  
  \  \  
   \
```

Notice that when the sequence of heights has $C + 1$ elements, the corresponding skyline has only C columns. Given a sequence of heights, you must draw the skyline according to the rules given above. See the examples for further clarification.

Input

The input contains several test cases, each one described in exactly two lines. The first line contains an integer C indicating the number of columns of the skyline ($1 \leq C \leq 70$). The second line contains $C + 1$ integers H_i separated by single spaces representing the sequence of heights ($0 \leq H_i \leq 30$ for $1 \leq i \leq C + 1$). You may assume that there exist an height $H_i = 0$, and that the difference between successive heights is at most 1 (i.e. $|H_i - H_{i+1}| \leq 1$ for $1 \leq i \leq C$). The last line of the input contains a single -1 and should not be processed as a test case.

Output

For each test case output the corresponding skyline, followed by a line with exactly three asterisks (“***”). While writing the skyline, use only regular spaces, newlines, and the three characters mentioned in the statement. The skyline must be left aligned and it must contain exactly C columns. There must be no trailing spaces at the end of printed lines, neither empty lines.

Example

```
Input:  
8  
1 2 3 2 3 3 2 1 0  
3  
1 0 0 1
```

-1

Output:

```
  _  
 / \  
/   \  
 \   \  
  _
```

```
 \ /  
  _  
***
```