The purpose of this problem is to gain familiarity with stacks and queues.
You have three jugs that can hold c1, c2, and c3 liters of water, respectively. Initially, jug 1 is full and the other two jugs are empty. You can repeat the following procedure any number of times: Choose two of the jugs and pour the contents of one into the other until either the first is empty or the second is full. Your goal is to end up with exactly d liters in one of the jugs.
Write a program called WaterTransfer.java to determine the transfers required to reach the goal. The input is a single line containing four integers between 2 and 100 (inclusive) representing c1, c2, c3, and d. The output is a minimal sequence of jug contents, starting with the initial contents and ending with one of the jugs containing d liters. Each line of the output should consist of 3 integers separated by spaces. If no solution exists, then your program should produce no output.
Good test case: 10 5 3 4 ; from movie ”Die hard: with a vengeance”.
For example, if the input is

```
20 5 3 4
```

then a valid output is

```
20 0 0
15 5 0
15 2 3
18 2 0
18 0 2
13 5 2
13 4 3
```

There may be other solutions, but none with fewer transfers (6) than this one.