CS211

Non-recursive solution for Towers of Hanoi

Using the algorithm discussed in class, write an iterative program
to solve the Towers of Hanoi problem.

The problem:

You are given three towers a, b, and c. We start with n rings on tower a
and we need to transfer them to tower b subject to the following
restrictions:

1. We can only move one ring at a time, and
2. We may never put a larger numbered ring on top of a smaller numbered
one.

There are always 3 towers. Your program will prompt the user for the
number of rings.

Here is the algorithm.

Definition: A ring is "available" if it is on the top of one of the towers.

Definition: The "candidate" is the smallest available ring that has not been moved on the most recent move. The first candidate is ring 1.

The Algorithm:

1. Find the candidate.
2. Move the candidate (right or left, depending if the number of rings is odd or even) to the closest tower on which it can be placed. Move "around the circle" if necessary.
3. If not done, go back to step 1.

The output should be a set of "commands" of the following form:

"Move ring x from tower y to tower z" for each move.

In addition, your program should take 2^n-1 moves for any n.