The Josephus Problem

The Josephus problem is the following game. There are *n* people, numbered 1 to *n*, sitting in a circle. Starting at person 1, a hot potato is passed. After *m* passes, the person holding the hot potato is eliminated, the circle closes ranks, and the game continues with the person who was sitting after the eliminated person picking up the hot potato. The last remaining person wins.

The Josephus problem arose in first century A.D., in a cave on a mountain in Israel, where Jewish zealots were besieged by Roman soldiers. The zealots voted to form a suicide pact rather than surrender to the Romans. Josephus suggested the game, and rigged the game so he would get the last lot. That is how we know about this game; in effect Josephus cheated!

Create an application that uses the circular list to model the problem. Download the CircApp.java file from course website and complete the main program.

Inputs are the number of people in the circle, the number used for counting off, and the number of the person where counting starts (usually 1). The output is the list of persons being eliminated. When a person drops out of the circle, counting starts again from the person who was on his left (assuming you go around clockwise).

Example:

There are 7 people numbered 1 through7, and you start at 1 and count off by three. People will be eliminated in the order 4, 1, 6, 5, 7, 3. Number 2 will be left!