Faculty of Engineering, University of Jaffna

Department of Computer Engineering

EC4070: Data Structures and Algorithms

Lab – 04

Linear Abstract Data Types

Date: 15th March 2022 **Duration: 3 hours**

1. Priority Queue - Maximize array sum after k negations [40 Marks]

Given an array A[ ] of size n and a number k, you can be able to modify the array k number of times. Here modifying array means in each operation you can replace any array element A[i] by -A[i]. You need to perform this operation in such a way that after k operations, sum of the array must be maximum.

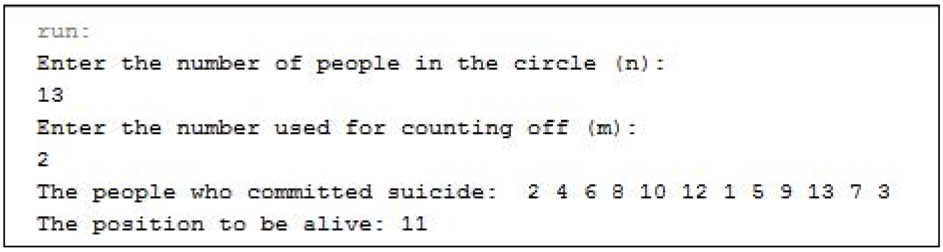
2. Circular Linked List [60 Marks]

The Josephus Problem is a famous mathematical puzzle that goes back to ancient times. There are many stories to go with the puzzle. One is that Josephus was one of a group of Jews and the group **(number of people in group is n)** was about to be captured by the Romans. Rather than be enslaved, Jews chose to commit suicide. They arranged themselves in a circle and, starting at a certain person, started counting off around the circle. Every **m**th person **(the counting off number is m)** had to leave the circle and commit suicide. When a person drops out of the circle, the counting starts again from the person who was on his right (assuming you go around clockwise).

Josephus decided that he didn't want to die, so he arranged the rules so he would be the last person left. The problem is made much more complicated because the circle shrinks as the counting continues. Implement a Java program that uses a circular linked list to model this problem.

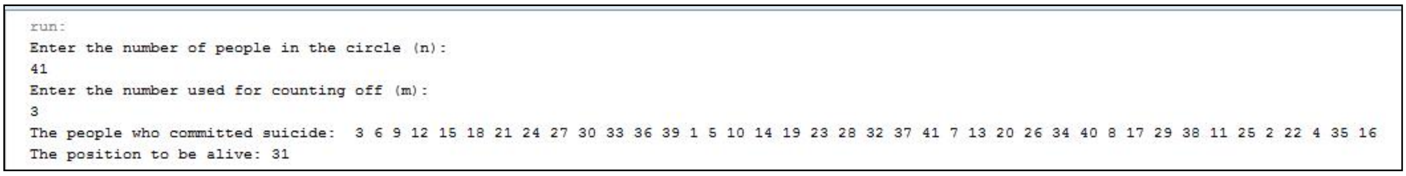
- Inputs are the number of people in the circle (n) and the number used for counting off (m).
- Here we assume that the counting starts with the first person.
- The output is
  - the people who committed suicide at each execution / count off **in the order**.
  - the preferred position for Josephus to be alive

Test case - 1:

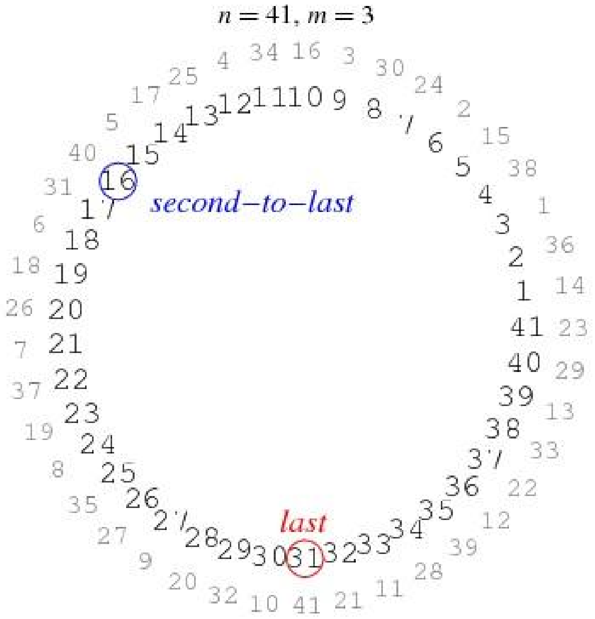


```
run:
Enter the number of people in the circle (n):
13
Enter the number used for counting off (m):
2
The people who committed suicide:  2 4 6 8 10 12 1 5 9 13 7 3
The position to be alive: 11
```

Test case - 2:



```
run:
Enter the number of people in the circle (n):
41
Enter the number used for counting off (m):
3
The people who committed suicide:  3 6 9 12 15 18 21 24 27 30 33 36 39 1 5 10 14 19 23 28 32 37 41 7 13 20 26 34 40 8 17 29 38 11 25 2 22 4 35 16
The position to be alive: 31
```



The original Josephus problem consisted of a circle of 41 men with every third man killed ($n = 41$, $m = 3$), illustrated above, where the outer number indicates the order in which a given man is killed. In order for the lives of the last two men to be spared, they must be placed at positions 31 (last) and 16 (second-to-last). The complete list in order of execution is 3, 6, 9, 12, 15, 18, 21, 24, 27, 30, 33, 36, 39, 1, 5, 10, 14, 19, 23, 28, 32, 37, 41, 7, 13, 20, 26, 34, 40, 8, 17, 29, 38, 11, 25, 2, 22, 4, 35, 16, 31.

Instructions:

- Implement Java programs for given questions using best coding practices. You should name your classes using appropriate names.
- Create a zip file named 201x_E_xxx_L4 which contains the Java programs for question1 and 2, and upload the zip file on/before given deadline via team.
- Any plagiarized work will end up in 0 marks.