

COP 3502C Summer 2019
Midterm Assignment # 2
Total point: 13

Introduction: For this assignment you have to write a c program that will use the concept of circular queue using circular doubly linked list.

What should you submit?

Write all the code in a single file and upload the .c file in webcourses.

Please include the following commented lines in the beginning of your code to declare your authorship of the code:

/* COP 3502C Midterm Assignment One
This program is written by: Your Full Name */

Compliance with Rules: UCF Golden rules apply towards this assignment and submission. Assignment rules mentioned in syllabus, are also applied in this submission. The TA and Instructor can call any students for explaining any part of the code in order to better assess your authorship and for further clarification if needed.

Problem

In this assignment, you have to simulate the Josephus problem. There are n number of prisoners standing in a circle waiting to be executed. The counting out begins at some point in the circle and proceeds around the circle in a fixed direction. In each step, a certain number of people are skipped and the next person is executed. The elimination proceeds around the circle (which is becoming smaller and smaller as the executed people are removed), until only the last person remains, who is given freedom.

Given the total number of persons n and a number k which indicates that k-1 persons are skipped and kth person is killed in circle. The task is to choose the place in the initial circle so that you are the last one remaining and so survive.

Example

For example, if $n = 5$ and $k = 2$, then the safe position is 3. Firstly, the person at position 2 is killed, then person at position 4 is killed, then person at position 1 is killed. Finally, the person at position 5 is killed. So, the person at position 3 survives.

If $n = 7$ and $k = 3$, then the safe position is 4. The persons at positions 3, 6, 2, 7, 5, 1 are killed in order, and person at position 4 survives.

Input:

n and k

Output:

The position number who will survive.

Requirement:

You must use the concept of circular queue while inserting and it should be implemented using circular doubly linked list. You don't need to use front and rear as you are using linked list. Root and last node automatically handle that. There is no use of doubly part of the linked list though. However, this part is mainly to exercise the implementation of doubly linked list.

Your code must compile in EUSTIS server. If it does not compile in Eustis server, we conclude that your code does not compile even if it works in your computer.

Hint: After getting the value of n,

- generate n numbers (1, 2, 3., ..., n) and insert into the doubly circular linked list.
- Then start deleting nodes based on the value of k until the list has only one node remaining.
- How would you know that you have only one node? Maybe if head->next is head? Or maybe use a counter variable to keep track? There can be many ways to know that.
-

Rubric:

- 1) If code does not compile: 0
- 2) Use of doubly linked list: 2 point
- 3) Use of circular linked list: 5 point
- 4) Incorrect test result per test case: -2 points

Please see the lecture slides and uploaded codes for learning doubly linked lists.