COP 3502C
Midterm Assignment # 2
Total point: 12

Introduction: For this assignment you have to write a c program that will use the 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:

<p style="color:green">/* COP 3502C Midterm Assignment Two
This program is written by: Your Full Name */</p>

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

X-Kingdom has trapped n number of soldiers of their opponent. They want to execute them. They found out a strategy so that the prisoners will kill each other and at the end one prisoner will be alive and he will be released. As part of the process, they labeled each trapped soldier a sequence number starting from 1 and ending at n. So, all the n number of prisoners standing in a circle waiting to be executed. However, one soldier was distracting the sequence and it was found out that they were standing in reverse order instead of proper order like the following (lets say n = 7):

7 -> 6-> 5-> 4 -> 3 -> 2-> 1
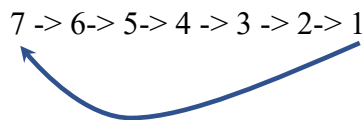
After realizing the wrong order of sequence, they reversed the circle to the correct order:

1 ->2-> 3-> 4 -> 5 -> 6-> 7
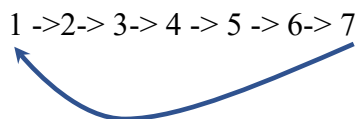
Now they will start the executing. The counting out begins at some point in the circle and proceeds around the circle in a fixed direction. In each step, a certain number of people are skipped and the next person is executed. The elimination proceeds around the circle (which is becoming smaller and smaller 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:

Print the list of prisoners in reverse order from n to 1

Then reverse the list to print it in correct order from 1 to n

Display the position number who will survive.

Requirement:

You must have to use circular doubly linked list for your solution. You need to declare appropriate doubly linked list node structure to store a soldier with sequence number as the data value. In addition to the other functions, you code must have to implement the following functions as use them part of the solution:

    a. soldier* create_soldier (int sequence): It takes an integer, dynamically allocate a soldier structure and returns a soldier node

b. soldier* create_reverse_circle(int n): It takes an integer and create a circular doubly linked list with n number of soldiers placed them in descending sequence. For example, if n=5 it should produce a circular doubly linked list starting from 5 and ending at 1 as sequence number.
After creating the circle, it returns the head of the circular linked list.

c. soldier* rearrange_circle(soldier* head): This function reverse a given circular doubly linked list where head is the head pointer. After reversing the linked list, it returns the head pointer of the updated linked list

d. void display(soldier* head): This function display a given circular doubly linked list. head is head pointer of the linked list

e. int kill(soldier* head, int n, int k): This function takes head of the linked list, number of soldiers n, and skip value k as parameter and returns the sequence number of the survived soldier. So, this function is kind of perform the killing task based on specification discussed above.

Note that in addition to the above functions, you will utilize other linked list functions for insertion, deletion, and you may create more function as you need for your solution.

During the killing process, there is no use of doubly part of the linked list. 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,

- It is always a good idea to plan it and do some paper work to understand the problem.
- generate n numbers (n, ...3, 2, 1) in reverse order and insert into the doubly circular linked list.
- The reverse the doubly circular linked list. One trick would be start from head and sequentially swap prev and next to opposite direction.
- 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: you will get 0

2) If you are missing any specified function: you will get 0

3) Not using doubly linked list: -2 points

4) Not using circular linked list: -2 points

5) Generating linked list with n sequence in the reverse order n…..3 2 1: -2

6) Re-arranging the linked list to correct order 1 2 3 .. n: -2

7) Incorrect test result per test case: -1 points (4 test cases)

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