Data Structure Homework C: Linked Lists

TA: Nick (weihan9453@gmail.com)

Deadline: 2017/11/14 11:59pm

1. Implement Polynomial Arithmetic

This homework's objective is to use the **link list** to solve the Josephus Problem.

2. Josephus Problem

Josephus problem is a problem where there are people 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 specified direction. At 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. The illustration of the processing about Josephus problem is shown in Figure 1. Assume there are eight people. Each number represents a person. K variable is used as a specified number of people who are skipped. Therefore, in the first round, number two is killed. The procedure is repeated with the remaining people, starting with the next person, going in the same direction and skipping the same number of people, until only one person remains, and is survived.

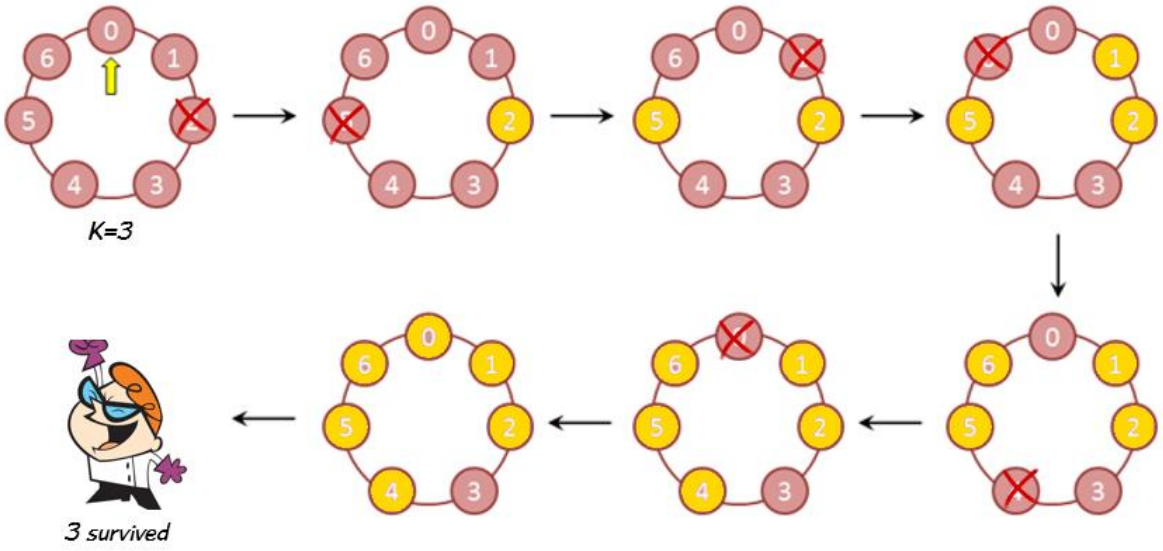


Figure 1. An Illustration of the Josephus Problem.

3. Implementation

3.1 Read data from a file

Your program should read the input file and correctly display values with circular linked lists. Besides, please add an action menu so that the user can choose (1) reading numbers from a file or (2) inputting new numbers. Figure 2 is an example for an action menu. If you choose (1) *Read numbers from a file*, the program reads a txt file directly and print out the numbers. After that, the program requires the user to input the number of persons to be skipped and a direction (left or right) to start the algorithm. On the other hand, if you choose (2) *Inputting new numbers*, the program requires the user to input the number of persons to be added. Figure3 shows the above-mentioned processes.

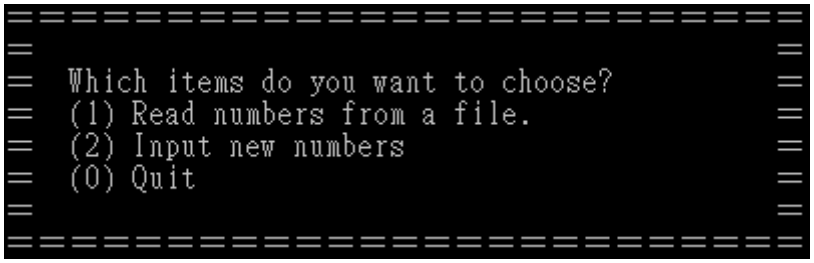
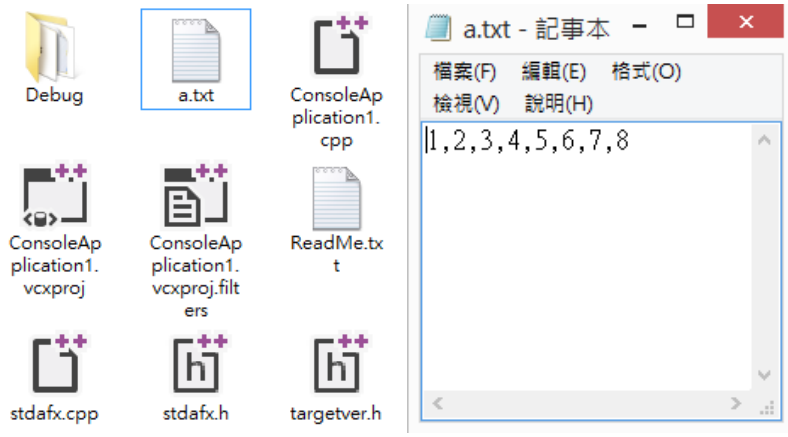
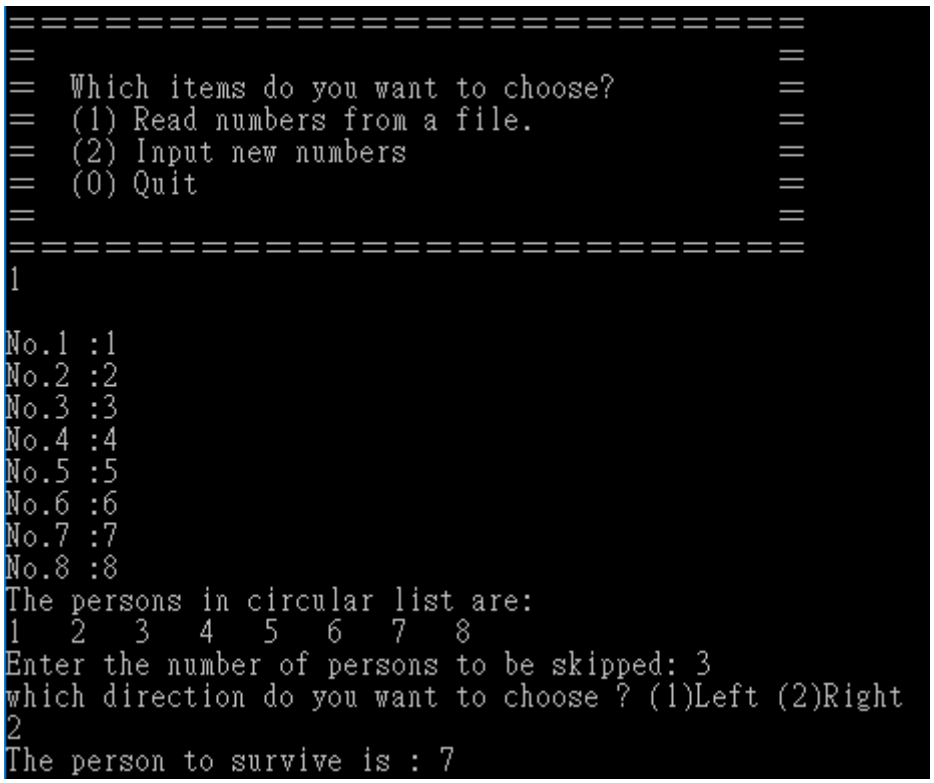


Figure 2. An Illustration of an action menu.





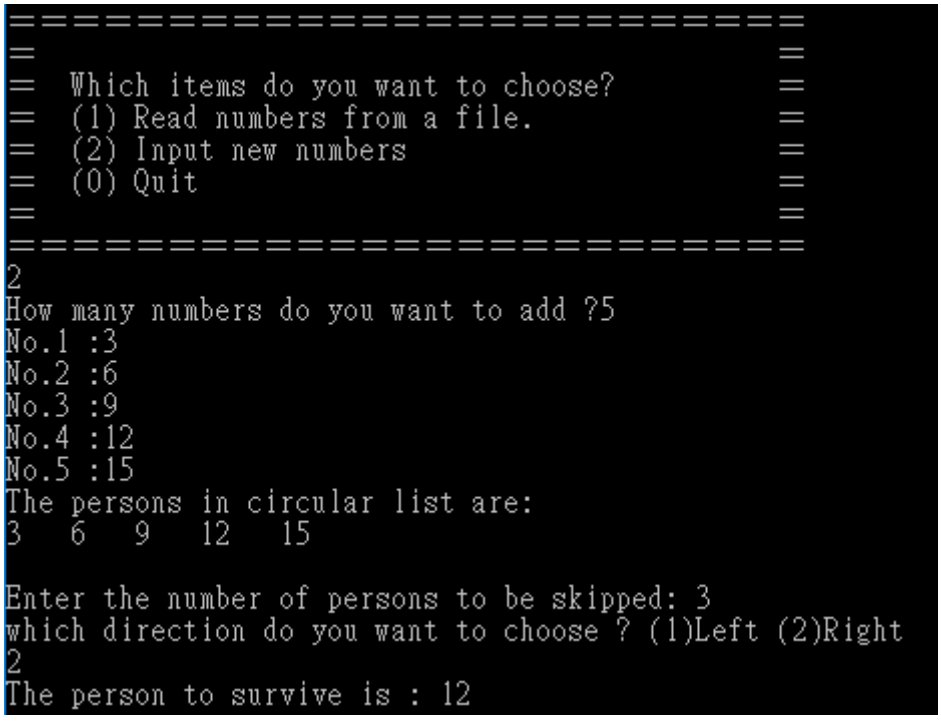


Figure 3. An Illustration of the program processes.

3.2 Circular Linked List

A circular linked list is a variation of a linked list in which the first element points to the last element and the last element points to the first element. There is no NULL at the end. Both a singly linked list and a doubly linked list can be used to make a circular linked list.

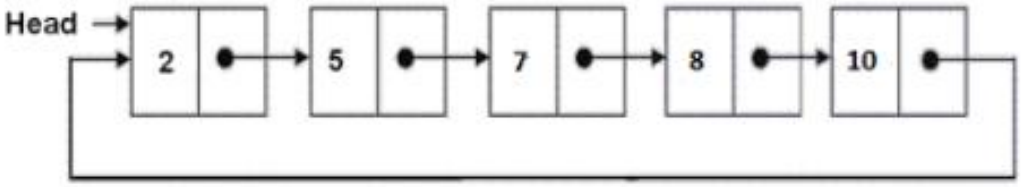


Figure 4. An illustration of circular linked list.

Please implement a program which creates a circular linked list and displays the elements in the list. Finally, use it to solve the Josephus problem.

4 Test case examples

Sample Input:	**Sample Output:**
1 (read a txt file) Skipped: 3 2 (right)	1 2 3 4 5 6 7 8 7
1 (read a txt file) Skipped: 3 1 (left)	1 2 3 4 5 6 7 8 3
2 (input numbers) 5 3 6 9 12 15 Skipped: 3 2 (right)	3 6 9 12 15 12
2 (input numbers) 10 7 4 1 0 2 5 8 9 6 3 Skipped: 5 1(left)	7 4 1 0 2 5 8 9 6 3 6

5 Submit

To submit your files electronically, enter the following command from the csie workstation:

turnin ds.hw3 [your files…]

To check the files you turnin, enter the following command from the csie workstation:

turnin –ls ds.hw3

You can see other description about turnin from following link:

https://www.cs.ccu.edu.tw/~lab401/doku.php?id=turninhowto

6 Grade policies

The TA(s) will mark and give points according to the following rules:

5% - Action menu.
10% - Read file and input new numbers.
80% - Circular Linked list and Josephus problem.
5% - Readme file, coding style, and comments in the source code.

- A document named “readme.txt”. You should describe the details of your project in your readme file in **English**.