Lab 11: Circular Queues

Aim

This lab class gives you an opportunity to:

- experiment with doubly-linked lists; and
- explore circular queues.

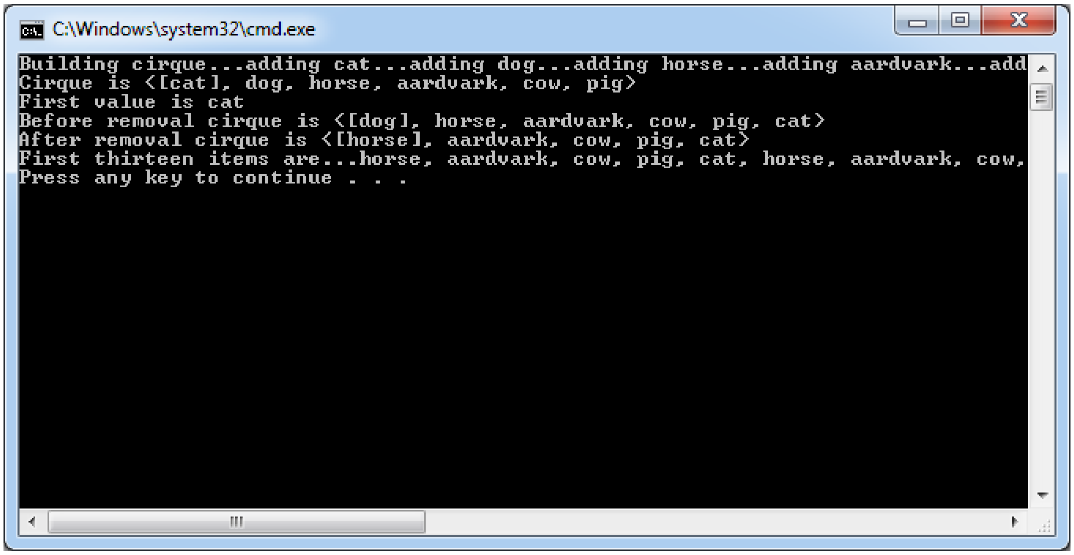
Context

Circular queues are queues in which there is no logical beginning or end. When implemented as doubly-linked-lists the 'last' node's `next` field should refer to the 'first' node and the 'first' node's `prev` field should refer to the 'last' node. A *cursor* can be used to refer to the 'selected' item of the circular queue. Insertions to the circular queue would occur prior to the cursor.

Tasks

1. Download the ZIPped folder (`Lab11.zip`) from MyLO to your home directory and extract all the files. Within this extracted folder is a project folder entitled `Lab11`. Open this project folder and open the project file (`Lab11.sln`).
3. Complete the implementation by writing the bodies of all functions in `cirque.c`. `nextOne()` should return the 'selected' item and select the next item in the circular queue; `rear()` should remove the 'selected' item, and then selects the *next* item, and `add()` should insert a new item *before* the selected item (leaving the selection unchanged).

Your implementation should be complete when you see the following output for the included driver file:



```
Building cirque...adding cat...adding dog...adding horse...adding aardvark...add
Cirque is <[cat], dog, horse, aardvark, cow, pig>
First value is cat
Before removal cirque is <[dog], horse, aardvark, cow, pig, cat>
After removal cirque is <[horse], aardvark, cow, pig, cat>
First thirteen items are...horse, aardvark, cow, pig, cat, horse, aardvark, cow,
Press any key to continue . . .
```