

CS 260: Arrays

Revision Date: January 10, 2013

Topics to learn

Please read up on the following subjects:

- dynamic arrays
- circular arrays
- public interfaces

Tasks to perform

Implement the following classes and following operations:

- a circular array class
 - get
 - set
 - insert-at-front
 - insert-at-back
 - remove-from-index
 - remove-from-front
 - remove-from-back
 - extract
- a dynamic array class
 - get
 - set
 - insert-at-back
 - remove-from-back

- remove-from-index
 - extract
- a dynamic circular array class
 - get
 - set
 - insert-at-front
 - insert-at-back
 - remove-from-front
 - remove-from-back
 - remove-from-index
 - extract

The extract operation returns a simple array containing just the current input data at their logical locations with no unoccupied array slots.

In addition to the operations listed above, all classes should implement *is-empty* and *is-full* (if appropriate) as well as *display*.

Each class should be implemented in a separate file. Provide a main function that thoroughly tests the interfaces of each class in a way that your Mentor can easily ascertain the correctness of your solution.

Other requirements

Be able to describe to your mentor the complexity of each operation in the public interface of your classes using order notation.

Note: an operation should be implemented as efficiently (order-notation-wise) as possible. Do not spend time optimizing an operation unless directed.