Circular Array Queue Tips

- ArrayQueue wraps around, remember when incrementing field variables:
  - frontIndex = (frontIndex + 1) % contents.length
  - backIndex = (backIndex + 1) % contents.length
- ArrayQueue can have one unused location to distinguish different states (full vs empty), or a size field variable could be used
- EmptyQueueException class can be used to handle cases when methods are called on an empty queue

One Unused Location Relationship

- Empty Queue:
  - frontIndex == (backIndex + 1) % contents.length
- Full Queue:
  - frontIndex == (backIndex + 2) % contents.length
- Queue with one entry:
  - frontIndex == backIndex

Queue Implementations
New Strategies

- To make use of the space in an array, we implement a queue with a **circular array** that wraps around from the last slot to the first
- To efficiently enqueue with a linked chain implementation of a queue a **lastNode** reference can be used and maintained
- To efficiently removeBack from a deque a **doubly linked chain** can be used and maintained