CSCA48 Exercise 3

Due: June 2, 2017. 5:00pm

A Neck Queue (NeQueue?)

Last week in lecture, we created a NeckStack: a stack where the 2nd item from the top is taken if available instead of the top item.

Your exercise for this week is to create a NeckQueue: a queue where the 2nd item in the queue is dequeued if there are 2 or more items. If there's only 1 item, it should be dequeued. All other operations behave as normal).

Your NeckQueue class should inherit from Queue (which will be provided by us in a separate file called container.py). Queue in turn will inherit from Container.

You should try to leave as much work as possible up to the Queue and Container classes. Be careful though, you don't know how either Container or Queue are implemented. So you can't assume anything about their internal workings.

The Queue and Container ADT

To make sure we're all on the same page, the docstrings for Container and Queue are given below:

```
class Container():

   def __init__(self):
       '''(Container) -> NoneType
       Create a new empty container
       '''

   def put(self, new_item):
       '''(Container, obj) -> NoneType
       Add new_item to this container
       RAISES: ContainerFullError if this container can't hold any more items
       '''

   def get(self):
       '''(Container) -> obj
       Remove and return an object (order not guaranteed) from this container
       RAISES: ContainerEmptyError if this container is empty
       '''

   def is_empty(self):
       '''(Container) -> bool
       Return True iff this container is empty
       '''
```

```
class Queue(Container)
   def __init__(self):
       '''(Queue) -> NoneType
       Create a new empty queue
       '''

   def is_empty(self):
       '''(Queue) -> bool
       Return True iff this queue is empty
       '''

   def enqueue(self, item):
       '''(Queue, object) -> NoneType
       Put item into the rear of the queue
       '''

   def dequeue(self):
       '''(Queue) -> object
       Remove and return the element at the front of the queue
       RAISES: QueueEmptyError if the queue is empty
       '''
```

What to hand in

Write your `NeckQueue` class (and no other classes/global code) in a file called `ex3.py`. The file should start with the line `from container import *`, which will import the `Queue` and `Container` classes, along with any other classes that may be in the file.