Name: ______________________________ Andrew Id: ______________________________

15-121 Fall 2018 Quiz 6

Up to 20 minutes. No calculators, no notes, no books, no computers. Show your work!
There are questions on *both sides* of this paper.

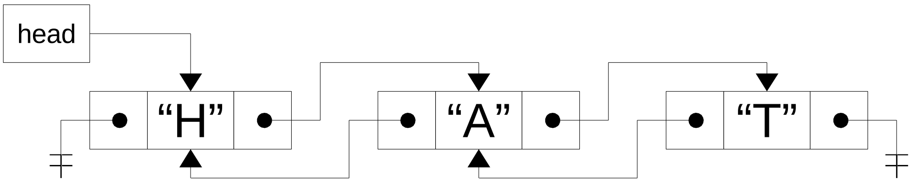
1. (8 points) Interfaces

   Consider a new abstract data structure called a `RandomQueue`. Like a regular queue, it allows you to enqueue and dequeue data, but it also has an additional method that dequeues a random item from the queue. A `RandomQueue` only needs to provide these three pieces of functionality.

   Write an interface for `RandomQueue`. You can choose the method names yourself, but you should make sensible choices. Note that you are only writing an interface.

2. (12 points) Linked Lists

In a doubly-linked list, each node in the list contains a pointer to both the next node in the list and the previous node in the list. In diagram form, a doubly-linked list would look like:



A class for this could be implemented as:

```
public class DoublyLinkedList<DataType> {
    private ListNode head = null;

    private class ListNode {
        public DataType data;
        public ListNode prev = null;
        public ListNode next = null;

        public ListNode(DataType value) {
            this.data = value;
        }
    }

    public void add(DataType value) {
        if (head == null) {
            // Add to an empty list
            ListNode newNode = new ListNode(value);
            newNode.next = head;
            if (head != null) {
                head.prev = newNode;
            }
            newNode.prev = null;
            head = newNode;
        } else {
            // Add to the end of the list
            ListNode tmpNode = head;
            while (tmpNode.next != null) {
                tmpNode = tmpNode.next;
            }
            tmpNode.next = new ListNode(value);
            tmpNode.next.prev = tmpNode;
        }
    }

    public boolean remove(DataType value) {
        // You will write this code.
    }
}
```

Write the code for the remove method, which removes the first node in the list containing `value`. It returns `true` if a node was removed and `false` if `value` was not found.

Write your answer on the next page.

```
public boolean remove(DataType value) {
```