

# CSC 212 Tutorial #4

## List & DLL

### Problem 1

Write the method *removeBetween*, member of the class *DoubleLinkedList*. The method takes two elements  $e_1$  and  $e_2$ , and removes all the elements between the two elements ( $e_1$  and  $e_2$  not included). If  $e_1$  or  $e_2$  or both doesn't exist, no element will be removed. You can assume the elements to be unique, and that  $e_1 \neq e_2$ . **Do not call any methods and do not use any auxiliary data structures.** The method signature is: `public void removeBetween(T e1, T e2)`.

**Example 1.1.** Given the list:  $A \leftrightarrow B \leftrightarrow C \leftrightarrow D \leftrightarrow E \leftrightarrow F$ , `removeBetween('B', 'E')` results in:  $A \leftrightarrow B \leftrightarrow E \leftrightarrow F$ .

### Problem 2

Write the method *reverseCopy*, user of *DoubleLinkedList*, which copies the elements of  $l1$  to  $l2$  in reverse order. The list  $l1$  must not change. Assume that  $l2$  is empty. The method signature is `public static <T> void reverseCopy(DoubleLinkedList<T> l1, DoubleLinkedList<T> l2)`.

**Example 2.1.** If  $l1 : A \leftrightarrow B \leftrightarrow C \leftrightarrow D$ , then calling `reverseCopy(l1, l2)` results in  $l2 : D \leftrightarrow C \leftrightarrow B \leftrightarrow A$ .