CS 311 Worksheet 4A Key

Questions are based on the following declarations:

```
public class DListNode
{
        private String data;
        private DListNode next;
        private DListNode back;
        public DListNode();
        ... ...
}

public class DList
{
      private DListNode top;
      private DListNode end;
      private int length;
      public DList();
      public boolean isEmpty();
      public void insertTop(String item);
      public void insertEnd(String item);
      public String deleteTop();
      public String deleteEnd();
      public void Print();
      public void revPrint();
      public void sort();     // to ascending order
      public boolean isPresent(String item);
      public void deleteItem(String item);
      // If item is found, delete it; otherwise, send a message
}
```

Write the implementations for the following: i) default constructor; ii) insertTop; iii) deleteEnd; iv) deleteItem

i)

```
      public DList()
      {     length = 0;
            top = null;
            end = null;
      }
```

ii)

```
      public void insertTop(String item)
      {     DListNode temp = new DListNode(item, null, null);
            if(isEmpty())
            {     top = temp;
                  end = temp;
            }
            else
            {     temp.setNext(top);
                  top.setBack(temp);
                  top = temp;
            }
            length ++;
      }
```

iii)

```
      public void deleteItem(String item)
      {     boolean find = false;
            DListNode temp = top;
            while(temp != null)
            {     if(temp.getData().equalsIgnoreCase(item))
                  {     find = true;
                        if(top == end)    // only one node
                        {     top = null;
                              end = null;
                        }
                        else              // two or more
                        {     if(temp == top)
                              {     top = top.getNext();
                                    top.setBack(null);
                              }
                              else if(temp == end)
                              {     end = end.getBack();
                                    end.setNext(null);
                              }
                              else
                              {     DListNode last = temp.getBack();
                                    DListNode next = temp.getNext();
                                    last.setNext(next);
                                    next.setBack(last);
                              }
                              System.out.println("\n" + item + " was
removed.");
                        }
                  }
                  // move one node forward
                  temp = temp.getNext();
            }
            if(!find)   System.out.println("\n" + item + " was not found.");
      }
```