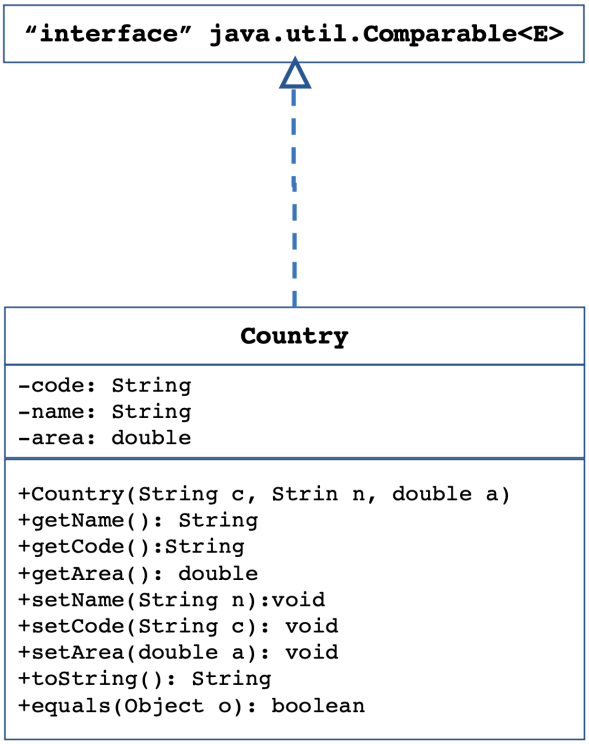
Programming and Data Structures
Test #2

You are provided with a Java program completed at 75% only. Your task is to complete the missing parts as described below.

The program uses the class **`Country`** as shown in the UML diagram below. The class is provided to you in the file **`Country.java`**.



The program has four parts:

1. The program creates three lists of type **`Country`**. The first list is an array list, the second is a linked list and the third is a stack. The three lists are populated with the data read from the file `countries.txt`. The method to read the data from the file is provided to you (you do not need to make any changes to it). An array of type **`List`** is used to save the references of the three lists. All the manipulations on the three lists use only the common methods from the interface **`List`**.

2. The program searches for a country name in the three lists using a generic search method that returns the index at which the country is found or -1 otherwise. Write the definition of the function `search` with the signature below. The method must be **generic** and implemented using **recursion**.

```
public static <E> int search(List<E> list, E key)
```

3. The program sorts the three lists using a comparator object to sort the countries by name and then sorts the same lists using a comparator object to sort the countries by area. Write the definition of the sort method and create two classes for the two types of comparators (comparator by name and comparator by area). Use the selectionSort algorithm provided at the end of this document and modify it to accept a list and a comparator object. The signature of the method is provided below. The sort method uses the comparator object **c** to compare the elements of **list**.

```
public static <E> void sort(List<E> list, Comparator<E> c)
```

Note that the three calls to the method **sort** (lines 40, 48, and 58 in the class **Test**) are commented out to avoid compiler errors. Make sure you uncomment these lines once you have defined the comparator classes.

4. Finally, the program combines the three lists in one list and sorts the final list. The program uses the method **combineAll** as described below. Write the definition of the method **combineAll** , which must be **generic** and use an **iterator**. The method adds all the elements from **list2** to **list1**.

```
public static <E> void combineAll(List<E> list1, List<E> list2)
```

5. Determine the time complexity of the methods **search**, **sort**, and **combineAll** using Big-O notation. Write the big-O notation as a comment before the method headers.

Important note: You are not allowed to use any method from the interface **`List`**, other than the methods listed below:

Method from `List<E>`	**Description**
`Iterator<E> iterator()`	Returns an iterator object positioned at the head of the list You can call **`next()`** and **`hasNext()`** on the returned object
`boolean add(E element)`	Adds **`element`** to the list
`int size()`	Returns the number of elements in the list
`E get(int index)`	Returns the value of the element at **`index`**
`E set(int index, E newV)`	Modifies the value of the element at **`index`** to **`newV`**, and returns the old value of the element
`boolean contains(E elem)`	Returns **`true`** if **`elem`** is in the list, **`false`** otherwise

Do not modify the class **`Country`** or the methods **`readData`**, **`printList`**, and **`main`** in the class **`Test`**.

Use the reference output of the program provided below for testing.

Submit all the Java files on coursesite.

`Important note: Javadoc` comments are not required.

Expected output of the program

```
------------------------------------------------------------
List 1:
[
(AO, Angola, 1246700.0sq.ft)
(HR, Croatia, 55974.0sq.ft)
(GT, Guatemala, 107159.0sq.ft)
(KZ, Kazakhstan, 2699700.0sq.ft)
(CT, Central African Republic, 622984.0sq.ft)
(MJ, Montenegro, 13452.0sq.ft)
(SU, Sudan, 1861484.0sq.ft)
]
List 2:
[
(BR, Brazil, 8358140.0sq.ft)
(ES, El Salvador, 20721.0sq.ft)
(TN, Tonga, 717.0sq.ft)
(LT, Lesotho, 30355.0sq.ft)
(NG, Niger, 1266700.0sq.ft)
(SL, Sierra Leone, 71620.0sq.ft)
(BG, Bangladesh, 130170.0sq.ft)
]
List 3:
[
(ST, Saint Lucia, 606.0sq.ft)
(GB, Gabon, 257667.0sq.ft)
(ID, Indonesia, 1811569.0sq.ft)
(US, United States, 9148655.0sq.ft)
(MV, Maldives, 298.0sq.ft)
(RP, Philippines, 298170.0sq.ft)
]
United States not found in list 1
United States not found in list 2
United States found in list 3 at index: 4

Sorting the lists by name
List 1:
[
(AO, Angola, 1246700.0sq.ft)
(CT, Central African Republic, 622984.0sq.ft)
(HR, Croatia, 55974.0sq.ft)
```

```
(GT, Guatemala, 107159.0sq.ft)
(KZ, Kazakhstan, 2699700.0sq.ft)
(MJ, Montenegro, 13452.0sq.ft)
(SU, Sudan, 1861484.0sq.ft)
]
List 2:
[
(BG, Bangladesh, 130170.0sq.ft)
(BR, Brazil, 8358140.0sq.ft)
(ES, El Salvador, 20721.0sq.ft)
(LT, Lesotho, 30355.0sq.ft)
(NG, Niger, 1266700.0sq.ft)
(SL, Sierra Leone, 71620.0sq.ft)
(TN, Tonga, 717.0sq.ft)
]
List 3:
[
(GB, Gabon, 257667.0sq.ft)
(ID, Indonesia, 1811569.0sq.ft)
(MV, Maldives, 298.0sq.ft)
(RP, Philippines, 298170.0sq.ft)
(ST, Saint Lucia, 606.0sq.ft)
(US, United States, 9148655.0sq.ft)
]


Sorting the lists by area
Sorted List 1:
[
(MJ, Montenegro, 13452.0sq.ft)
(HR, Croatia, 55974.0sq.ft)
(GT, Guatemala, 107159.0sq.ft)
(CT, Central African Republic, 622984.0sq.ft)
(AO, Angola, 1246700.0sq.ft)
(SU, Sudan, 1861484.0sq.ft)
(KZ, Kazakhstan, 2699700.0sq.ft)
]
Sorted List 2:
[
(TN, Tonga, 717.0sq.ft)
(ES, El Salvador, 20721.0sq.ft)
```

```
(LT, Lesotho, 30355.0sq.ft)
(SL, Sierra Leone, 71620.0sq.ft)
(BG, Bangladesh, 130170.0sq.ft)
(NG, Niger, 1266700.0sq.ft)
(BR, Brazil, 8358140.0sq.ft)
]
Sorted List 3:
[
(MV, Maldives, 298.0sq.ft)
(ST, Saint Lucia, 606.0sq.ft)
(GB, Gabon, 257667.0sq.ft)
(RP, Philippines, 298170.0sq.ft)
(ID, Indonesia, 1811569.0sq.ft)
(US, United States, 9148655.0sq.ft)
]

All lists combined and sorted by area:
[
(MV, Maldives, 298.0sq.ft)
(ST, Saint Lucia, 606.0sq.ft)
(TN, Tonga, 717.0sq.ft)
(MJ, Montenegro, 13452.0sq.ft)
(ES, El Salvador, 20721.0sq.ft)
(LT, Lesotho, 30355.0sq.ft)
(HR, Croatia, 55974.0sq.ft)
(SL, Sierra Leone, 71620.0sq.ft)
(GT, Guatemala, 107159.0sq.ft)
(BG, Bangladesh, 130170.0sq.ft)
(GB, Gabon, 257667.0sq.ft)
(RP, Philippines, 298170.0sq.ft)
(CT, Central African Republic, 622984.0sq.ft)
(AO, Angola, 1246700.0sq.ft)
(NG, Niger, 1266700.0sq.ft)
(ID, Indonesia, 1811569.0sq.ft)
(SU, Sudan, 1861484.0sq.ft)
(KZ, Kazakhstan, 2699700.0sq.ft)
(BR, Brazil, 8358140.0sq.ft)
(US, United States, 9148655.0sq.ft)
]
```

Generic Selection Sort Method

```
public static <E extends Comparable<E>> void sort(E[] list){
     int currentMinIndex;
     E currentMin;
     for (int i=0; i<list.length-1; i++) {
          currentMinIndex = i;
          currentMin = list[i];
          for(int j=i+1; j<list.length; j++) {
               if(currentMin.compareTo(list[j]) > 0) {
                    currentMin = list[j];
                    currentMinIndex = j;
               }
          }
          list[currentMinIndex] = list[i];
          list[i] = currentMin;
     }
}
```