Sofia University
Department of Mathematics and Informatics

Course : OO Programming Java
Date: November 27, 2018
Student Name:

Lab No. 8a

Submit the all Java files developed to solve the problems listed below. Use comments and Modified-Hungarian notation.

Използвайте изцяло средствата на Collections библиотеката за решаване на следните задачи

Problem 1

P1. Write class **`ArrayListTest`** to test the following methods.

a) Write a generic method that returns the **maximum element** in **a two-dimensional array**.

```
public static <E extends Comparable<E>> E max(E[][] list)
```

b) Write the following method **that shuffles** an **`ArrayList`**:

```
public static <E> void shuffle(ArrayList<E> list)
```

c) Write the following method that **returns the largest element** in an **`ArrayList`**:

```
public static <E extends Comparable<E>> E max(ArrayList<E> list)
```

d) Write the following method that returns a new **`ArrayList`**. The new list contains the **non-duplicate** elements from the original list.

```
public static <E> ArrayList<E> removeDuplicates(ArrayList<E> list)
```

Problem 2

2.1 Modify Program **`WordTypeCount`** (`fig19_20.rar` use the attached code) **from lecture 8c** to output the results sorted in **descending order** of the values.

2.2 Write class **`ArrayListTest`** that has an **`ArrayList`** with elements of type **`String`**.

a) Write the following method

```
public void sort()
```

for sorting the **`ArrayList`** elements in descending order. Use a **`Comparator`** instance.

b) Write the following method

```
public void sortByFrwequency()
```

for sorting the **`ArrayList`** elements in descending order of the frequency of the elements. Use a **`Comparator`** instance.

c) Write the following method

```
public static <T extends Comparable<? super T>>
                                                T removeMax(List<T> list)
{ ... }
```

to deletes the element with the largest value from list and returns a reference to the value. If the **`list`** is empty **`removeMax`**() returns **`null`**

d) Write the following method

`public void getNames()`

to read from standard input the names of students and store them in **`ArrayList`** . **Filter and store** only unique names in the **`ArrayList`**.

e) Write the following method

`public void searchNames()`

that searches for a student name stored in the **`ArrayList`** and outputs a message in case the name is found or not found

f) Write the following method

`public void copyTo(ArrayList<String> str)`

that copies **`ArrayList`** elements into **`ArrayList<String> str`**

g) Write the following method

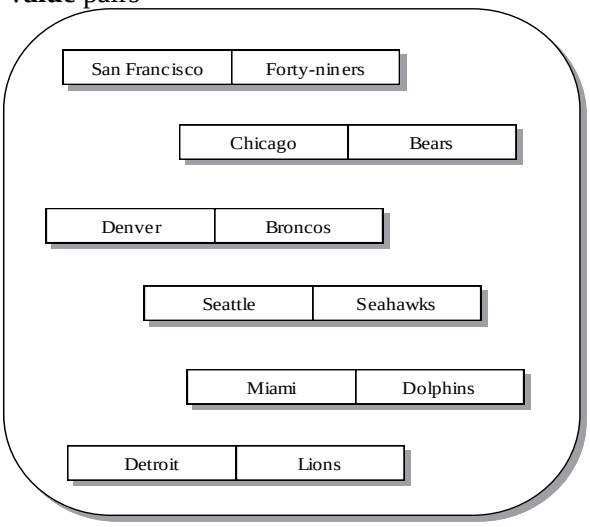
`public String toStringDescending()`

that returns a String with the elements of **`ArrayList`** in descending order

Test the above methods in the **`main`**() memethod of class **`ArrayListTest`**

Задача 3

3.1 Write a program that creates a **`TreeMap<String, String> map`**, containing the following **key-value** pairs



Perform the following actions with the map.

- Output the size of the map and the name of the team in **`Chicago`**.
- Change the name of the **`San Francisco`** team to "**`Niners`**".
- Output whether **`San`** Diego has a team in the map.
- Remove **`Denver`** from the map.

- Insert the **`Dallas Cowboys`** in the map.
- Using an **entry set `iterator`**, scan the map and remove all teams in a city beginning with the letters 'M'-'Z'.

Output the final map

3.2 Using an iterator, implement a static version of the method **`addAll()`** that forms **the union of a set and elements in a collection** and returns the result as a Set.

```
public static <T> Set<T> addAll(Set<T> s, Collection<? extends T> c)
{ . . . }
```

The wildcard notation **`"? extends T`**" refers to any type that is a subtype of T. In this situation,

```
? "is a" T
```

If you use an iterator, **`iter`**, to traverse **`Collection c,`** declare it as follows:

```
Iterator<? extends T> iter = c.iterator();
```

In a program, create a **`TreeSet<String>`** with elements from **`String`** array **`strArrA`** and an **`ArrayList`** with elements from array **`strArrB`**. Use **`addAll`**() to create a **`TreeSet`** that is a union of the set and array list. Output the results.

```
String strArrA[] = {"dog", "cat", "tiger", "pig"},
strArrB[] = {"frog", "dog", "monkey", "pig", "snake"};
```

Problem No. 4.

Write a program that **finds the most frequently occurring element** in an array of integers. Use `ArrayLists` to count the number of occurrences of each element. At the end print the element and its number of occurrences. If there is more than one such element, any one of them may be printed. When the program is done, we will see what needs to be changed to make it work on an array of Strings.
The starting code is below.

```
import java.util.ArrayList;

public class MostFrequentElement {

    /**
       Given an array of ints, the program finds and prints the most
       frequently occurring element and its number of occurrences.
       If there is more than one such element, any one of them may be
       printed.
       Assume that the given array contains at least one element.
     **/
    public static void main(String [] args) {
// in this example 1 is the most frequent element,
// it appears 7 times:
        int [] elements = {1, 3, 4, 1, 5, 2, 3, 6, 6, 6, 4, 1, 2, 6,
                           2, 3, 1, 2, 1, 5, 5, 1, 1, 5, 4};
```

```
        // Your code goes


        // fill in the appropriate results:
        System.out.println("The most frequent element " +  " occurs " +  "
times");
    }
}
```

Problem 5

Напишете програма, която има методи, за да :

a) **създава `ArrayList`**, в който се **записва всяко изречение в отделен елемент на списъка**., а после **създава втори списък** който е копие на първия, но елементите му са записани в обратен ред .

Тествайте приложението като изведете на стандартен изход елементите на двата списъка, а също и **броя на елементите** от всеки списък