University of Bahrain

College of Information Technology

Department of Computer Science

First Semester, 2018-2019

**ITCS214 / ITCS215 / ITCS216 (Data Structures)**

**Assignment 2**

It is required to develop an application in Java to represent Set as a data structure using Java built-in class **ArrayList**.

1. Write a class called **Set** to represent a set as an **ArrayList** and implement various set operations by using methods of class **ArrayList**. Assume that the elements of the set are integers.

This class will have the following instance variables: (a) an **ArrayList** object called **list** of type **Integer**, (b) **length**: the actual number of elements in the **list**.

This class will have the following methods:

1. Constructor without any parameter (default constructor), which uses a default value of 10 as **capacity**, creates **ArrayList** object **list** and initializes **length** to 0.
2. Constructor with parameter **cap** for **capacity** of the **list**. Create **ArrayList** object **list** of capacity equal to parameter **cap** and initialize **length** to 0.
3. Constructor having an array of integers **arr** as parameter. Create **ArrayList** object **list.** Add theelements of the array **arr** in the **list**. and initialize **length** to the length of the array **arr**.
4. Instance method **getLength**, that returns **length**.
5. Instance method **isEmpty** to determine whether the set is empty or not.
6. Instance method **addElement** that accepts a parameter **element** of type **int**. It will insert the **element** in the **list** at the end and also increment **length** by one.
7. Instance method **isMember** that accepts a parameter **element** of type **int**. If **element** exists in the **list**, it will return true, else it will return false.
8. Instance method **subSet** that accepts a parameter **aSet** of type **Set.**  If aSet is a subset of “this object”, it will return true, else it will return false.

Ex: Let A and B be two sets;

A = {7, 4, 20, 15, 12}, B = {12, 15}. As all elements of B are contained in A, therefore, B is a subset of A.

1. Instance method **isEqual** that accepts a parameter **aSet** of type **Set.**  The method will return true, if **aSet** is equal to “this object”, else it will return false. Two sets are equal, if they contain the same elements in any order.

Ex: Let A and B be two sets;

A = {7, 4, 20, 15, 12}, B = {4, 7, 12, 15, 20}. As sets A and B contain same elements, but in different order, therefore, they are equal.

1. Instance method **union** having a parameter **aSet** of type **Set**. The method finds the union of **aSet** with “this object” and returns the result as an object of type **Set**.

Ex: Let A = {10, 4, 20, 15, 12, 18}, B = {8, 10, 25, 15, 20},

C = A ∪ B = {10, 4, 20, 15, 12, 18, 8, 25}

1. Instance method **intersection** having a parameter **aSet** of type **Set**. The method finds the intersection of **aSet** with “this object” and returns the result as an object of type **Set**.

Ex: Let A = {10, 4, 20, 15, 12, 18}, B = {8, 10, 25, 15, 20},

C = A ∩ B = {10, 20, 15}.

1. Write a class called **SetApplication** having only **main** method. Create objects of class type **Set** using different constructors and call methods of class **Set** to test all functionalities of class **Set**.