

WIA1002/WIB1002 Data Structure**Lab : Priority Queue**

1. Given an integer array which consists of {4, 8, 1, 2, 9, 6, 3, 7}. Insert these integers into a priority queue using its ADT. Then, perform the following operations to the priority queue:
 - toString() - Display all the elements inside this priority queue.
 - poll() – retrieve and remove the first element in this priority queue.
 - add() – add new element 5 into the priority queue
 - toArray() – convert the priority queue into an array and display.
 - peek() – retrieve the first element in the priority queue.
 - contains() – check if the priority queue consists of element “1”.
 - size() – get the current size of the priority queue.
 - isEmpty() – display while removing the elements in the queue until it is empty.
 - Sort the priority queue in reversing order.

2. Create two priority queues with the following elements: {"George", "Jim", "John", "Blake", "Kevin", "Michael"} and {"George", "Katie", "Kevin", "Michelle", "Ryan"}. Find their union, difference, and intersection.

3. Given following books information and the main class:

```
import java.util.Queue;
public class TestComparableBook {
    public static void main(String[] args) {
        Queue<ComparableBook> BookQueue = new java.util.PriorityQueue<>();
        BookQueue.add(new ComparableBook(1065, "Effective Java: Third Edition"));
        BookQueue.add(new ComparableBook(3012, "Java: A Beginner Guide Seventh Edition"));
        BookQueue.add(new ComparableBook(1097, "Learn Java in One Day and Learn It Well"));
        BookQueue.add(new ComparableBook(7063, "Beginning Programming with Java
(Dummies)"));
        BookQueue.add(new ComparableBook(6481, "Java: Programming Basic for Absolute
Beginner"));

        System.out.println(BookQueue);
        while (BookQueue.peek() != null) {
            System.out.println("Head Element: " + BookQueue.peek());
            BookQueue.remove();
            System.out.println("Priority queue: " + BookQueue);
        }
    }
}
```

Write the code for ComparableBook class using Comparable.