# EC4070: DATA STRUCTURES AND ALGORITHMS

# LAB 08: GREEDY ALGORITHMS

**NAME** : **WIJAYAWARDHANA W.A.H.A.**

**REGISTRATION NO.** : **2019/E/166**

**DATE ASSIGNED** : **26 MAY 2022**

Code:-

```
import java.util.Scanner;
/**
 * import Scanner class.
 */

/**
 * Creating a java class as {@link BottleFilling2019E166} for getting the maximum number of bottles which can fill from the container.
 */


public class BottleFilling2019E166 {
   Scanner scanner = new Scanner(System.in);
   int numberOfBottles;
   int capacityOfTheContainer;

   /**
    * numberOfBottles taken from the user for entering the number of bottles which he need to fill from the main container.
    * capacityOfTheContainer means the total water which provided at the beginning for filling to the other small containers.
    * An array to store the capacity of each container which are willing to fill.
    */

   /**
    * settingBigContainer method use to taken the values of the numberOfBottles and capacityOfTheContainer form the user.
    */

   public void settingBigContainer()
   {
      System.out.println("Enter number of bottles and capacity of container : ");
      numberOfBottles = scanner.nextInt();
      capacityOfTheContainer = scanner.nextInt();
   }



   int capacityOfBottle[];
   /**
    * settingCapacityOfBottle method use to input the capacity of each bottle which are willing to fill from the big container.
    */

   public void settingCapacityOfBottle()
   {
```

```
        int capacityOfBottle01[] = new int[numberOfBottles];
        System.out.println("Enter bottle capacities : ");
        for(int i =0; i< numberOfBottles; i++)
        {
            int tempValue = scanner.nextInt();
            capacityOfBottle01[i] = tempValue;
        }
        capacityOfBottle =  capacityOfBottle01;
    }

    /**
     *settingAscendingOrderOfBottles method use to make the containers according to the capacity of each.
     */


    public void settingAscendingOrderOfBottles()
    {
        for(int i = 0; i< numberOfBottles; i++)
        {
            int minimumCapacity = capacityOfBottle[i];
            int minimumIndex = i;
            int j = i+1;
            for(; j < numberOfBottles; j++)
            {
                if(minimumCapacity > capacityOfBottle[j])
                {
                    minimumCapacity = capacityOfBottle[j];
                    minimumIndex = j;
                }
            }
            int tempValue = capacityOfBottle[i];
            capacityOfBottle[i] = minimumCapacity;
            capacityOfBottle[minimumIndex] = tempValue;

        }
    }

    /**
     * fillingWaterContainer method use to fill the water from lowest capacity to highest capacity until maximum number of bottles fill from it.
     */

    public void fillingWaterContainer()
    {
        int numberOfWaterContainerFilled = 0;
        int totalFilledWaterCapacity = 0;
        int totalRemainingWaterCapacity = capacityOfTheContainer;
        while(totalRemainingWaterCapacity >= capacityOfBottle[numberOfWaterContainerFilled])
```

```
        {
            totalFilledWaterCapacity = capacityOfBottle[numberOfWaterContainerFilled];
            totalRemainingWaterCapacity=totalRemainingWaterCapacity-
capacityOfBottle[numberOfWaterContainerFilled];
            numberOfWaterContainerFilled++;

        }
        System.out.println("Maximum bottle can filled : "+numberOfWaterContainerFilled);
    }

    /**
     * main method use for creating the class object and calling the methods of the class.
     */

    public static void main(String[] args) {

        BottleFilling2019E166 bottle = new BottleFilling2019E166();
        bottle.settingBigContainer();
        bottle.settingCapacityOfBottle();
        bottle.settingAscendingOrderOfBottles();
        bottle.fillingWaterContainer();
    }
}
```

Output:-