ASSIGNMENT
01

NAME :- V. Bhavani

REG NO :- 192210198

COURSE CODE :- CSA0914

COURSE NAME :- Programming in Java For Raspberry Pi

Submitted To:-

Dr. R. Hemavathi

① pseudocode:-

```
PROGRAM ArrayList-operations
    DECLARE list AS ArrayList of String
  ADD "apple" to list
  ADD "Banana" to list
  ADD "cherry" to list
  ADD " dates" to list
  print "Initial List "+ list
  SET removeindex to 2
   Remove element at removeindex from list
  print "removed element " + removedelement
  print "List after removal:" + list.
  SET searchelement to "data"
  Set   searchindex to Index of searchelement in list
  if Searchindex is not -1 then
     print "element" + searchelement + " found at" + Searchindex
  else
     print " element not found"
  print "Iterating through the list:"
  for each element in list
   print element
  end for
end program.
```

output:-

```
Initial List: [apple, banana, cherry, dates]
removed element: cherry
element 'dates' found at 3.
```

program:-

```
import java.util.Arraylist;
public class Arraylistoperations {
    public static void main(String[] args) {
        ArrayList<String> list = new Arraylist <>();
        list.add ("apple"),
        list.add ("banana");
        list.add (" cherry");
        list.add ("dates");
      System.out.println("Initial list:" + list);
      int removeIndex = 2;
      String removedelement = list.remove (removeIndex);
      System.out.println ("removedelement:" +removedelement);
      System.out.println (" list after removal:" + list);
      String Searchelement = "date"
      if (Search.index ≠ -1) {
        System.out.println( "element found at:"+searchindex);
      }
     else {
        System.out.println (" element not found :");
      }
     System.out.println (" Iterating through the list:");
     for(String element : list) {
        System.out.println (element);
     }
   }
}
```

② pseudocode:-

```
program Hashsetoperations
  Declare nameset as Hashset of Strings
    Add "John" to nameset
    Add "Alice" to nameset
    Add "Bob" to nameset
    print "Initial set" + nameset.
    SET newname to "David"
    ADD newname to nameset.
    print "set after adding" + newname + ":" + nameset.
    SET removename to "Bob"
    Remove removename from nameset
    print "set after removing" + removename + ":" + nameset.
    SET Searchname to "Alice"
    If. nameset contains searchname then
      print "name is present in the set"
    else
      print "name is not present"
  end for
  end program.
```

output:-

```
Initial set:[Bob, John, Alice]
Set after adding david: [Bob, John, Alice, David]
set after removing Bob: [John, Alice, David]
Name 'Alice' is present in the set.
```

Program:-

```
import java.util.~~Scanner~~ Hashset;
public class Hashsetoperations{
  public static void main(String[]arg){
    Hashset<String>nameset =new Hashset<>();
      nameset.add("John");
      nameset.add("Alice");
      nameset.add("Bob");
     System.out.println("Initial set:"+ nameset);
      String newname= "David";
      nameset.add(newname));
     System.out.println("set after adding" +nameset);
      String removename ="Bob";
      nameset.remove(removename);
      System.out.println("set after removing" + nameset);
      String searchname= "Alice"
      if (nameset.contains(searchname)){
        System.out.println("name is found");
      }
      else {
        System.out.println("name is not present");
      }
      System.out.println("display all names:");
    }
  }.
```

③ Pseudocode:-

```
program priorityqueueExample
    Declare Employee As class
        Declare name as String
        Declare priority as Integer.
        Constructor Employee (name as String, priority as Integer)
            Set this.name to name
            Set this.priority to priority
        end Constructor
    end class
    Declare pq as priorityqueue of Employee
    Set pq to new priorityqueue ((e1, e2) => e2.priority - e1.priority)
    ADD new Employee ("John", 3) to pq
    ADD new Employee ("Alice", 1) to pq
    ADD new Employee ("Bob", 2) to pq
    ADD new Employee ("eve", 4) to pq.
    print "Initial priorityqueue:" + pq.
    SET highestpriorityEmployee to pq.poll()
end for
end program.
```

output:-

```
displaying priorityqueue:
Eve - priority: 4
John - priority: 3
Bob - priority: 2
Alice - priority: 1
```

program:-

```
import java.util.priorityqueue;
class Employee {
    String name;
    int priority;
     public Employee(Stringname, int priority) {
         this.name = name;
         this. priority = priority;
     }
  }
  public class priorityqueueExample {
    public Static void main(String() args) {
       priorityqueue<employee> pq = new priorityqueue<>
       pq.add(new Employee("John", 3));
      pq.add(new Employee("Alice", 1));
      pq.add( new employee ("Bob", 2));
      pq.add(new Employee ("eve", 4));
    System.out.println(" Initial priority " + pq);
    Employee highest priorityEmployee = pq.poll();
      System.out.println("removed employee " + highestpriority);
      System.out.println("priorityqueue after highest priority");
    }
  }
```

(4) pseudocode:-

```
program HashmapExample.
    Declare Studentmap As hashmap of integer to string.
     Add 101 to Studentmap with value "John".
     Add 102 to Studentmap with value "Alice".
     Add 103 to Studentmap with value "Bob".
     Add 104 to Studentmap with value "eve".
     print "Initial Hashmap:"+ Studentmap".
      Set searchid to 103
      If studentmap contain key searchid then
     print "Student id" + searchid+" Corresponds to "+ Studentmap)
    else
       print "Student id is not found".
    end if
    print "Hashmap after removing Student:"+Studentmap
    for each id in Studentmap. keyset
          print id"+id+ "name"+Studentmap. get(id)
   end for
 end program.
```

Output:-

```
Initial Hashmap: {101= John, 102 =Alice, 103=Bob ,104=Eve}
Student ID103 Corresponds to Bob.
~~Student~~
displaying all names:
ID: 101, Name: John
ID: 103, Name: Bob.
```

program:-

```
import java.util.Hashmap;
public class HashmapExample{
    public static void main(String[] args){
        Hashmap< Integer, String> Studentmap =new Hashmap<>();
        Studentmap.put (101, "John");
        Studentmap.put (102, "Alice");
        Studentmap.put (103, "Bob");
        Studentmap. put (104, "Eve");
        System.out.println("Initial Hashmap" + Studentmap);
        int Searchid = 103;
        if (Studentmap. Containskey (Searchid)){
            System.out.println(" name is present:");
        }
       else {
        System.out.println(" name is not present:");
       }
       System.out.println("Hash map after removing" + Studentmap);
    }
}
```