Project 1: Array and Array classes

1. Write a program that uses one dimensional array to store N integer numbers, the pr*ogram should perform many operations on the array, which are:*

1. Delete all occurrence of specific item in array.
   Example:
   Input : arr1[] = {1, 2, 3, 2, 5 } and item=2
   Output : 1 3 5
2. Add new item after all occurrence of given item.
   Example:
   Input : arr1[] = {1, 2, 3, 2, 5 } , new item= 66 and item=2
   Output : 1 2 66 3 2 66 5

3. Add item before all occurrence of given item.
   Example:
   Input : arr1[] = {1, 2, 3, 2, 5 } , new item= 66 and item=2
   Output : 1 66 2 3 66 2 5

4. Delete item in the middle of array.
   Examples:
   Input : arr1[] = {1, 2, 3, 2, 5 }
   Output : 1 2 2 5

   Input : arr1[] = {1, 2, 3, 2, 5,6 }
   Output : 1 2 5 6

5. Delete item after all occurrence of given item.
6. input two integer array and merge them in third array as in the following Example**:**

   Input : arr1[] = {1, 2, 3, 4, 5 }
   Input : arr2[] = {6, 7, 8, 9, 10}
   Output : 1 6 2 7 3 8 4 9 5 10

2. The Java library provides a Arrays class in the java.util package, It contains various methods for manipulating arrays.
The following steps shows the usage of Arrays class:
a. Declaration of array variable as:
type arrayName[ ]=new type[size];
b. Call any method in Array class as :
Arrays.methodName(parameters)

binarySearch(Type [ ]a, key)
This method searches the specified array a of type Type for the specified value (key), this method return index of the element key in array a if key is contained in the array; otherwise, (-(insertion point) - 1). The insertion point is defined as the point at which the key would be inserted into the array.

copyOf (Type [ ]arrayoriginal, int newLength)
This method returns a copy of the original array to new array, padded with zeros to obtain the specified length.

equals(Type [ ]a1, Type [ ]a2)
This method returns true if the two specified arrays a1 and a2 are equal to one another. (Type may be char, int, double, ...)

fill (Type [ ] a, val)
This method assigns the specified value val(same type of elements a) to each element of the specified array a. (Type may be char, int, double, ...)

sort(Type[ ] a)
This method sorts the specified array a of type Type into ascending (Type may be char, int, double, ...)

toString(Type[ ] a)
This method returns a string representation of the contents of the specified array a of Type(Type may be byte, char, double, float, int, ...

3. ArrayList Class

The Java library provides ArrayList in the java.util package.

Syntax:

ArrayList<E> ArrayListName = new ArrayList<E>();

where E is the type of elements(Integer, String, ...) held in the ArrayList
ArrayListName is the name of ArrayList.

The constructor ArrayList is used to create an empty list with an initial capacity to hold 10 elements.

Methods of ArrayList class: there are number of methods available which can be used directly using object of ArrayList class. in the following the important methods.

add(index, element) : Inserts the specified element at the specified position in this list. Shifts the element currently at that position (if any) and any subsequent elements to the right (adds one to their indices).

add(element): Appends the specified element to the end of this list.

remove(int index): Removes the element at the specified position in this list. Shifts any subsequent elements to the left.

set(index, element) : Replaces the element at the specified position in this list with the specified element.

get(index) : Returns the element at the specified position in this list.

indexOf(element) : Returns the index of the first occurrence of the specified element in this list, or -1 if this list does not contain the element.

contains(element): Returns true if this list contains the specified element, returns true if and only if this list contains at least one element .

lastIndexOf(elment): Returns the index of the last occurrence of the specified element in this list, or -1 if this list does not contain the element.

isEmpty() :Returns true if this list contains no elements.

size() : Returns the number of elements in this list.

clear(): Removes all of the elements from this list. The list will be empty after this call returns.