Arraylist

triangular braces

AL<Integer> arr = new AL<>();

arr.add(5); → 5

arr.add(6); → 5 | 6

arr.add(0,7); → 7 | 5 | 6

arr.set(1,2); → 7 | 2 | 6

arr.add(1,3); → 7 | 3 | 2 | 6

arr.remove(3); → 7 | 3 | 2

// arr.add(7,3); → error

arr.get(3); → error

```java
public static void main(String[] args) {
    ArrayList<Integer> arr = new ArrayList<>();

    Scanner scn = new Scanner(System.in);
    int t = scn.nextInt();
    for (int i = 0; i < t; i++) {
        int c = scn.nextInt();
        if (c == 1) {
            size(arr);
        } else if ( c == 2 ) {
            printAndRemoveFromLast(arr);
        } else if ( c == 3 ) {
            int x = scn.nextInt();
            printAndAddLast(arr, x);
        } else if ( c == 4 ) {
            printAndRemoveFromStart(arr);
        } else if ( c == 5 ) {
            int x = scn.nextInt();
            printAndAddAtStart(arr, x);
        } else if ( c == 6 ) {
            traverse(arr);
        }else {
            System.out.println("invalid");
        }
    }
}
```

1,2,3,4,5,6

```java
public static void size(ArrayList<Integer> arr) {
    int s = arr.size();
    System.out.println(s);
}

public static void printAndRemoveFromLast(ArrayList<Integer> arr) {
    if (arr.size() == 0) {
        System.out.println("invalid-move");
        return;
    }

    int ele = arr.get( arr.size() - 1 );
    System.out.println(ele);

    arr.remove( arr.size() - 1 );
}

public static void printAndAddLast(ArrayList<Integer> arr, int x) {
    System.out.println(x);
    arr.add(x);
}

public static void printAndRemoveFromStart(ArrayList<Integer> arr) {
    if (arr.size() == 0) {
        System.out.println("invalid-move");
        return;
    }

    int ele = arr.get(0);
    System.out.println(ele);

    arr.remove(0);
}

public static void printAndAddAtStart(ArrayList<Integer> arr, int x) {
    System.out.println(x);
    arr.add(0, x);
}

public static void traverse(ArrayList<Integer> arr) {
    if (arr.size() == 0) {
        System.out.println("invalid-move");
        return;
    }

    for (int i = 0; i < arr.size(); i++) {
        System.out.print( arr.get(i) + " " );
    }
    System.out.println();
}
```

ArrayList Printing

for and for each loop

```java
public static void main(String[] args) {
    ArrayList<Integer> arr = new ArrayList<>();

    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    for (int i = 0; i < n; i++) {
        int ele = scn.nextInt();
        arr.add( ele );
    }

    // using for loop
    for (int i = 0; i < arr.size(); i++) {
        System.out.print(arr.get(i) + " ");
    }
    System.out.println();

    // using for each loop
    for (Integer i : arr) {
        System.out.print(i + " ");
    }
}
```

here 'i' will be index

here 'i' will be AL value

ArrayList reverse printing

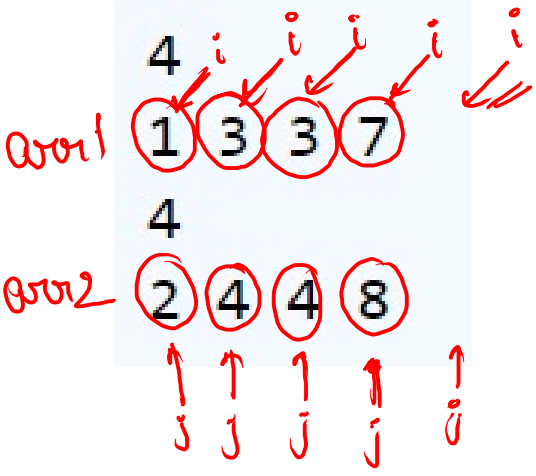
```java
public static void main(String[] args) {
    ArrayList<Integer> arr = new ArrayList<>();

    Scanner scn = new Scanner(System.in);
    int n = scn.nextInt();
    for (int i = 0; i < n; i++) {
        int ele = scn.nextInt();
        arr.add( ele );
    }

    // using for loop
    for (int i = arr.size() - 1; i >= 0; i--) {
        System.out.print(arr.get(i) + " ");
    }
    System.out.println();

    // using for each loop
    Collections.reverse(arr);   // O(N)
    for (Integer i : arr) {
        System.out.print(i + " ");
    }
}
```

// created arraylist

// for loop in reverse

// reverse AL and print

Merge two sorted arrays 7



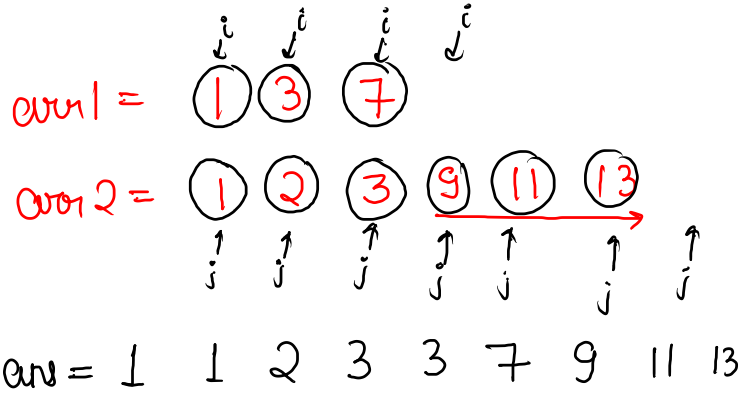

```
if ( arr1[i] < arr2[j] ) {
    ans.add ( arr1[i] );
    i++ ;
} else {
    ans.add ( arr2[j] );
    j++ ;
}
```

ans = 1 2 3 3 4 4 7 8

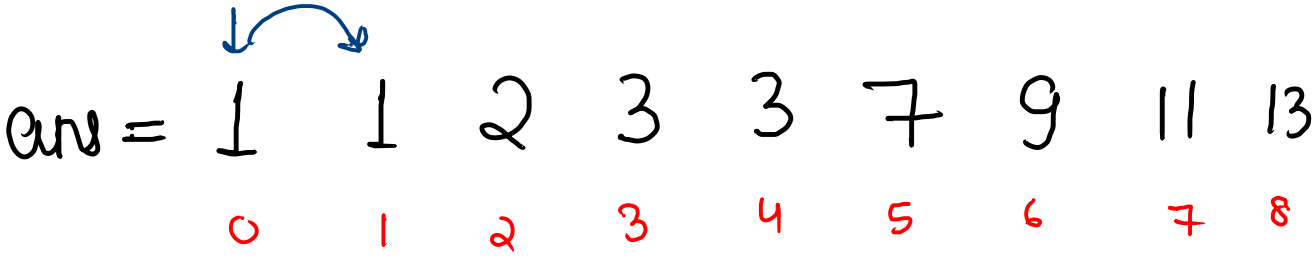
```java
int i = 0;
int j = 0;
while (i < arr1.length && j < arr2.length) {
    if ( arr1[i] < arr2[j] ) {
        ans.add( arr1[i] );
        i++;
    } else {
        ans.add( arr2[j] );
        j++;
    }
}
```

```java
while (i < arr1.length) {
    ans.add( arr1[i] );
    i++;
}
```

```java
while (j < arr2.length) {
    ans.add( arr2[j] );
    j++;
}
```



(right now, we are only trying merge 2 arrays into arraylist)




```
while( idx < ans.size()){
    if( ans.get(idx) == ans.get(idx+1)){
        ans.remove(idx);
    } else {
        idx++;
    }
}
```

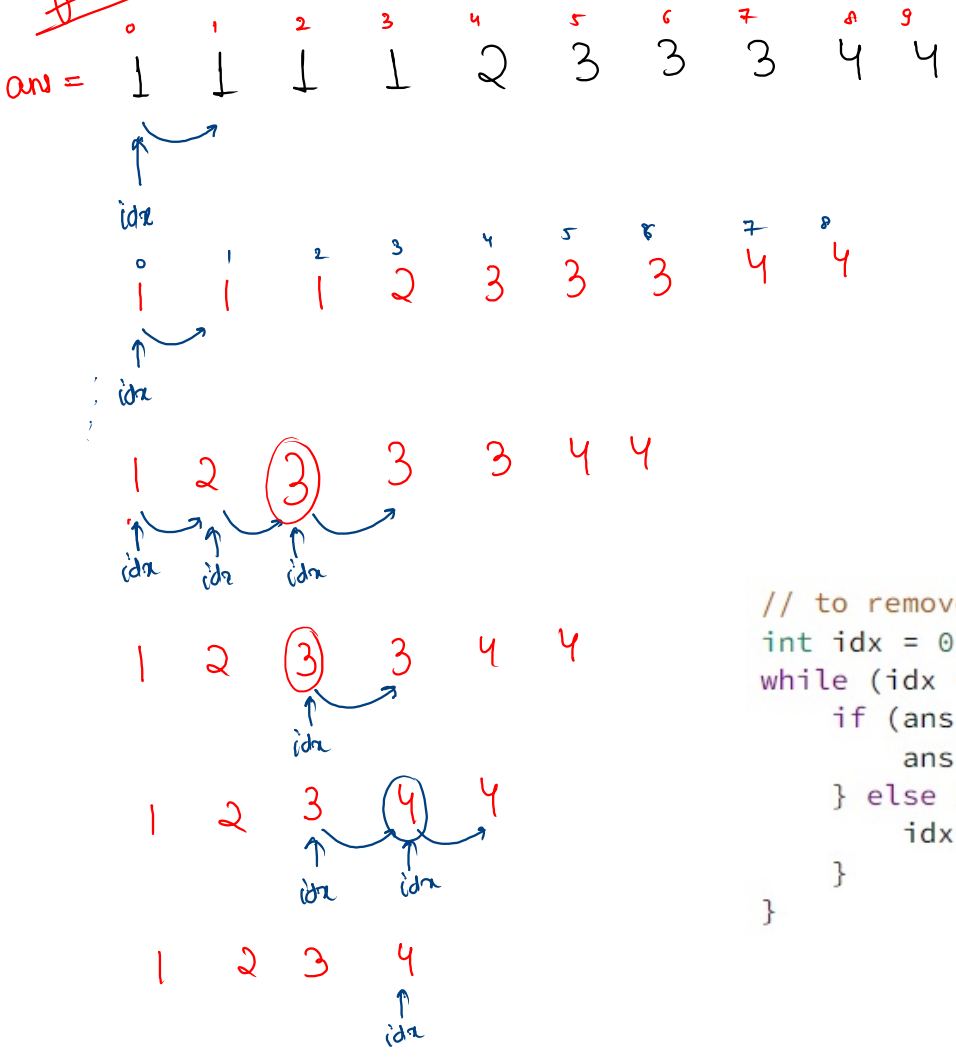
code

part 1

part 2

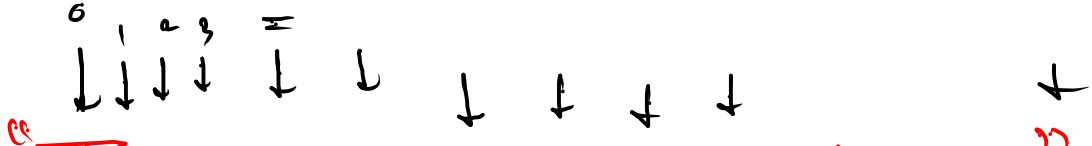
```java
public static void mergeArrays(int[] arr1, int[] arr2) {
    ArrayList<Integer> ans = new ArrayList<>();
    int i = 0;
    int j = 0;
    while (i < arr1.length && j < arr2.length) {
        if ( arr1[i] < arr2[j] ) {
            ans.add( arr1[i] );
            i++;
        } else {
            ans.add( arr2[j] );
            j++;
        }
    }

    while (i < arr1.length) {
        ans.add( arr1[i] );
        i++;
    }

    while (j < arr2.length) {
        ans.add( arr2[j] );
        j++;
    }

    // to remove duplicate
    int idx = 0;
    while (idx < ans.size() - 1) {
        if (ans.get(idx) == ans.get(idx + 1)) {
            ans.remove(idx);
        } else {
            idx++;
        }
    }


    // print
    for (Integer val : ans) {
        System.out.print(val + " ");
    }
}
```

dry run of part 2




```java
// to remove duplicate
int idx = 0;
while (idx < ans.size() - 1) {
    if (ans.get(idx) == ans.get(idx + 1)) {
        ans.remove(idx);
    } else {
        idx++;
    }
}
```

HW_reverse by words



str = ~~sihT~~ ~~si~~ a ecnetnes

ans = sihT_si_a_

ans = ans + str[i]

str = ch + str[i]

"sihT si a ecnetnes"