Group A

1. Given the following Java program:

```
import java.util.*;
public class Main
{
public static void main(String[] args) {
List num = new ArrayList(Arrays.asList(23, 16, 14, 33, 19, 6, 1));
System.out.println("List is "+num);
}
}
```

(a) Give the index values of all the odd numbers assuming zero-based indexing : 16, 33, 6
(b) How many elements would be looked at when the list is traversed (from start to finish) until the value 19 was found? : 4

2. Which of the following lists are syntactically correct in Java?

Try them out in to see if you were correct.

(a) List num = new ArrayList(Arrays.asList(1, 2, 3, 'four'));
(b) List num = new ArrayList(Arrays.asList(1, 2, [3, 4]));

```
import java.util.*;
public class Week4a2{
    public static void main(String[]args){
    List lA=new ArrayList();
    lA.add("1");
    lA.add("2");
    lA.add("3");
    lA.add("four");
    System.out.println("List is"+lA);
    }
}
```

BlueJ: Terminal Window - Week4-workshop

Options

```
List is[1, 2, 3, four]
```

3. Perform a series of list operations on the following list:

List fruit = new ArrayList (Arrays.asList('apple', 'banana', 'pear', 'cherry'));

to produce this updated list:

['Grapefruit', 'banana', 'Date', 'cherry', 'Orange']

```java
import java.util.*;
public class A3{
    public static void main(String[] args){
        List fruit=new ArrayList(Arrays.asList("apple", "Banana", "pear", "cherry"));
        fruit.add("Grapefruit");
        fruit.add("banana");
        fruit.add("Date");
        fruit.add("cherry");
        fruit.add("Orange");
        System.out.print(fruit);
    }
}
```

```
BlueJ: Terminal Window - workshp4
Options
[apple, Banana, pear, cherry, Grapefruit, banana, Date, cherry, Orange]
```

Group B

1. Write a program to find out whether a given integer is present in an array or not.

```java
import java.util.*;
public class Week4b1{
    public static void main (String[]args){
    int anyNumber=10;
    int[] a={1,2,3};
    if(a[0]==0 || a[1]==2|| a[2]==3){
         System.out.println("found");
        }
        else
        System.out.println("not found");
    }
}
```

```
BlueJ: Terminal Window - Week4-workshop
Options
found
```

2. Calculate the average marks from an array containing marks of all students in physics using a for-each loop.

```
public class Week4b2{
    public static void main(String[]args){
    int[] physics={2,9,0,5,12,22,33};
    int sum=0;
    Double average;
    for (int number: physics){
        sum+=number;
    }
    int arrayLength = physics.length;
    average =((double)sum /(double)arrayLength);
    System.out.println("The marks of all physics students");
    System.out.println("Sum="+sum);
    System.out.println("Average="+average);
     }
}
```

```
BlueJ: Terminal Window - Week4-workshop
Options
The marks of all physics students
Sum=83
Average=11.857142857142858
```

3. Write a Java program to reverse an array.

```
public class Week4b3{
    public static void main(String[] args){
        String[] ary={"Apple","Banana","Coconut","Durain","Eggplant"};
        int i=4;
        System.out.println("The reverse of Apple Banana Coconut Durain Eggplant
        while(i>=0){
            System.out.println(ary[i]);
            i--;
        }
    }
}
```

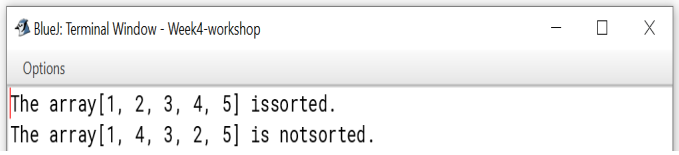
```
BlueJ: Terminal Window - Week4-workshop
Options
The reverse of Apple Banana Coconut Durain Eggplant is
Eggplant
Durain
Coconut
Banana
Apple
```

4. Write a Java program to find the maximum element in an array.

```java
public class Week4b4{
    public static void main(String[] args){
        int[] ary={2,5,12,15,20,1};
        int i=0;
        int max=ary[0];
        while(i<=5){
            int a= ary[i];
            if(max<a){
                max=a;
            }
            i++;
        }
        System.out.println("The max element is:"+max);
    }
}
```

5. Write a Java program to find whether an array is sorted or not.

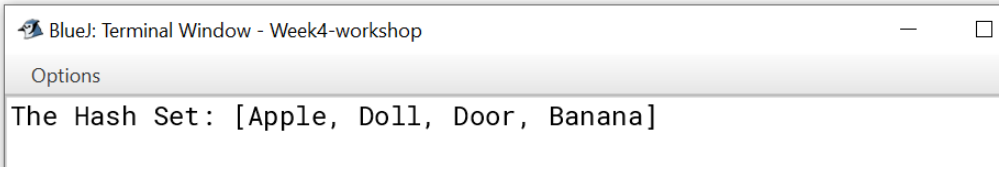
```java
import java.util.Arrays;
public class Week4b5{
private static boolean isSortedArray(int[] array, int n){
        if(n == 1 || n == 0)return true;
        return array[n-2] <= array[n-1]&& isSortedArray(array, n-1);  }
    public static void main(String[] args){
    int[] arr ={1,2,3,4,5};
    System.out.println("The array"+Arrays.toString(arr) +" "+(isSortedArray(arr, arr.length)?"is" :"is not") + "sorted.");
    arr = new int[]{1,4,3,2,5};
    System.out.println("The array"+Arrays.toString(arr) +" "+(isSortedArray(arr, arr.length)?"is" :"is not") + "sorted.");
}
}
```



Group C

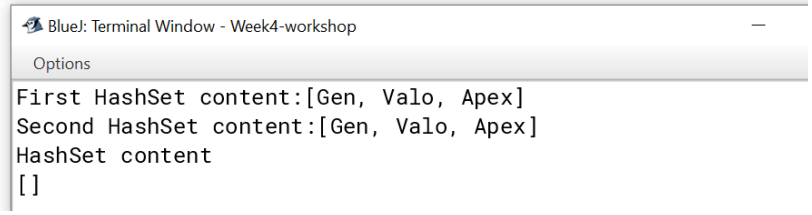
1. Write a Java program to append the specified element to the end of a hash set.

```java
import java.util.HashSet;
public class Week4c1 {
    public static void main(String[] args) {
    HashSet<String> haset = new HashSet<String>();
    haset.add("Apple");
    haset.add("Banana");
    haset.add("Doll");
    haset.add("Door");
    System.out.println("The Hash Set: "+ haset);
}
}
```



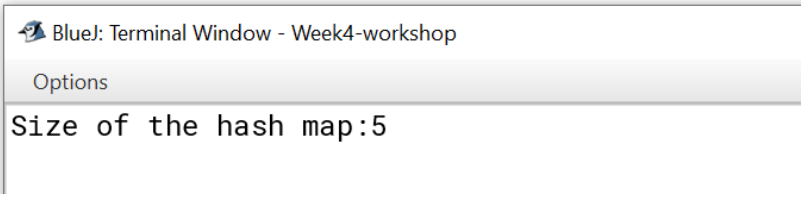
2. Write a Java program to compare two sets and retain elements which are same on both sets.

```java
import java.util.HashSet;
public class Week4c2  {
    public static void main(String[] args) {
        HashSet<String> hset = new HashSet<String>();
        hset.add("Apex");
        hset.add("Valo");
        hset.add("Gen");
        System.out.println("First HashSet content:"+hset);
        HashSet<String>hst  = new HashSet<String>();
        hset.add("Apex");
        hset.add("Valo");
        hset.add("Gen");
        System.out.println("Second HashSet content:"+hset);
        hset.retainAll(hst);
        System.out.println("HashSet content");
        System.out.println(hset);
    }
}
```



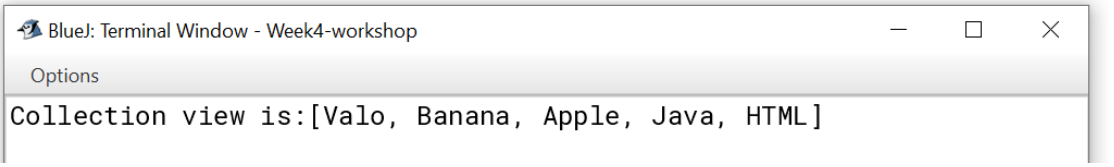
3. Write a Java program to count the number of key-value mappings in a hash table

```java
import java.util.*;
public class Week4c3 {
    public static void main (String[] args) {
        HashMap<Integer, String> map = new HashMap<Integer, String>();
        map.put(1,"Valo");
        map.put(2,"Banana");
        map.put(3,"Apple");
        map.put(4,"Java");
        map.put(5,"HTML");
        System.out.println("Size of the hash map:"+ map.size());
    }
}
```



4. Write a Java program to get a collection view of the values contained in this map

```java
import java.util.*;
public class Week4c4 {
    public static void main (String[] args){
        HashMap<Integer, String>hash_map = new HashMap<Integer, String>();
        hash_map.put(1,"Valo");
        hash_map.put(2,"Banana");
        hash_map.put(3,"Apple");
        hash_map.put(4,"Java");
        hash_map.put(5,"HTML");
        System.out.println("Collection view is:"+ hash_map.values());
    }
}
```



(Optional)

Group D

1. Building a Rock Paper Scissor game in java

```java
import java.util.Scanner;
public class RockPaperScissors
{
    public static void main(String[] args)
    {
        Scanner scan = new Scanner(System.in);
        System.out.println("Player 1: Choose rock, scissors, or paper:");
        String player1 = scan.next().toLowerCase();
        System.out.println("Player 2: Choose rock, scissors, or paper:");
        String player2 = scan.next().toLowerCase();

        if (player1.equals(player2))
        {
          System.out.print("It is a tie");
        }
    }
}
```