

| | |
|---------------------|-------------------------------------|
| Started on | Wednesday, 3 January 2024, 9:18 AM |
| State | Finished |
| Completed on | Wednesday, 3 January 2024, 10:34 AM |
| Time taken | 1 hour 15 mins |

Question **1**

Correct

Marked out of 25.00

Given an array `nums` containing `n` distinct numbers in the range `[0, n]`, return the only number in the range that is missing from the array.

Example 1:

Input:

3

3 0 1

Output: 2

Explanation: `n = 3` since there are 3 numbers, so all numbers are in the range `[0,3]`. 2 is the missing number in the range since it does not appear in `nums`.

Example 2:

Input:

2

0 1

Output: 2

Explanation: `n = 2` since there are 2 numbers, so all numbers are in the range `[0,2]`. 2 is the missing number in the range since it does not appear in `nums`.

Example 3:

Input:

9

9 6 4 2 3 5 7 0 1

Output: 8

Explanation: `n = 9` since there are 9 numbers, so all numbers are in the range `[0,9]`. 8 is the missing number in the range since it does not appear in `nums`.

Constraints:

`n == nums.length`

`1 <= n <= 10^4`

`0 <= nums[i] <= n`

All the numbers of `nums` are unique.

For example:

| Input | Result |
|------------------------|--------|
| 3 3 0 1 | 2 |
| 2 0 1 | 2 |
| 9 9 6 4 2 3 5 7 0 1 | 8 |

Answer: (penalty regime: 0 %)

```
1 import java.util.*;
2 public class UniqueNums
3 {
4     public static void main(String[] args)
5     {
6         Scanner sc = new Scanner(System.in);
7         int n = sc.nextInt();
8         ArrayList<Integer> a = new ArrayList<Integer>();
9         HashMap<Integer,Integer> hm = new HashMap<Integer,Integer>();
10        for(int i=0;i<n;i++)
11        {
12            a.add(sc.nextInt());
13        }
```

```

13     }
14     for(int i=0;i<=n;i++)
15     {
16         hm.put(i,Collections.frequency(a,i));
17     }
18     for(int i=0;i<=n;i++)
19     {
20         if(hm.getOrDefault(i,0)==0)
21         {
22             System.out.print(i);
23             break;
24         }
25     }
26 }
27 }

```

| | Input | Expected | Got | |
|---|------------------------|----------|-----|---|
| ✓ | 3 3 0 1 | 2 | 2 | ✓ |
| ✓ | 2 0 1 | 2 | 2 | ✓ |
| ✓ | 9 9 6 4 2 3 5 7 0 1 | 8 | 8 | ✓ |

Passed all tests! ✓

Question **2**

Correct

Marked out of 25.00

Given an integer array arr, count how many elements x there are, such that $x + 1$ is also in arr. If there are duplicates in arr, count them separately.

Example 1:

Input:

3

1 2 3

Output: 2

Explanation: 1 and 2 are counted cause 2 and 3 are in arr.

Example 2:

Input:

8

1 1 3 3 5 5 7 7

Output: 0

Explanation: No numbers are counted, cause there is no 2, 4, 6, or 8 in arr.

Constraints:

$1 \leq \text{arr.length} \leq 1000$

$0 \leq \text{arr}[i] \leq 1000$

For example:

| Input | Result |
|----------------------|--------|
| 3 1 2 3 | 2 |
| 8 1 1 3 3 5 5 7 7 | 0 |

Answer: (penalty regime: 0 %)

```
1 import java.util.*;
2 public class Count
3 {
4     public static void main(String[] args)
5     {
6         Scanner sc = new Scanner(System.in);
7         int n=sc.nextInt();
8         int c=0;
9         ArrayList<Integer> a=new ArrayList<Integer>();
10        for(int i=0;i<n;i++)
11        {
12            a.add(sc.nextInt());
13        }
14        HashMap<Integer,Integer> hm=new HashMap<Integer,Integer>();
15        for(int i=0;i<n;i++)
16        {
17            hm.put(i,a.get(i));
18        }
19        //System.out.print(hm);
20        for(int i=0;i<n;i++)
21        {
22            int val=hm.get(i);
23            val++;
24            if(hm.containsValue(val))
25            {
26                c+=1;
27            }
28        }
29        System.out.print(c);
30    }
31 }
```

```
31 |}
```

| | Input | Expected | Got | |
|---|----------------------|----------|-----|---|
| ✓ | 3 1 2 3 | 2 | 2 | ✓ |
| ✓ | 8 1 1 3 3 5 5 7 7 | 0 | 0 | ✓ |

Passed all tests! ✓

Question **3**

Correct

Marked out of 25.00

Create a map with name as key and roll number as value. Search for a name and replace it's value with a new value.

Input Format :

The first line of the input consists of the value of n.

Next input is the n names and roll numbers.

The third input is the key to be searched.

The fourth input is the value to be replaced.

Output Format :

The first line of the output prints the map with original values.

The next output prints the map with replaced values.

Sample testcases :

Testcase 1 Input

2

Alice

8

Mary

12

Alice

14

Testcase 1 Output

{Alice=8, Mary=12}

{Alice=14, Mary=12}

For example:

| Input | Result |
|-------|---------------------|
| 2 | {Alice=8, Mary=12} |
| Alice | {Alice=14, Mary=12} |
| 8 | |
| Mary | |
| 12 | |
| Alice | |
| 14 | |

Answer: (penalty regime: 0 %)

```
1 import java.util.*;
2 public class Replace
3 {
4     public static void main(String[] args)
5     {
6         Scanner sc = new Scanner(System.in);
7         int n=sc.nextInt();
8         HashMap<String,Integer> hm = new HashMap<String,Integer>();
9         for(int i=0;i<n;i++)
10        {
11            hm.put(sc.next(),sc.nextInt());
12        }
13        String s=sc.next();
14        int val=sc.nextInt();
15        System.out.println(hm);
16        hm.put(s,val);
17        System.out.print(hm);
18    }
19 }
```

| | Input | Expected | Got | |
|---|--|---|---|---|
| ✓ | 2 Alice 8 Mary 12 Alice 14 | {Alice=8, Mary=12} {Alice=14, Mary=12} | {Alice=8, Mary=12} {Alice=14, Mary=12} | ✓ |

Passed all tests! ✓

//

Question 4

Correct

Marked out of 25.00

Create a map with name as key and roll number as value. Search for a name and remove it.

Input Format :

The first line of the input consists of the value of n.

Next input is the n names and roll numbers.

The last input is the name to be removed.

Output Format :

The output prints the original list and the list after modification.

Sample testcases :

Testcase 1 Input

2

Alice

8

Mary

12

Alice

Testcase 1 Output

{Alice=8, Mary=12}

{Mary=12}

For example:

| Input | Result |
|-------|--------------------|
| 2 | {Alice=8, Mary=12} |
| Alice | {Mary=12} |
| 8 | |
| Mary | |
| 12 | |
| Alice | |

Answer: (penalty regime: 0 %)

```
1 import java.util.*;
2 public class Remove
3 {
4     public static void main(String[] args)
5     {
6         Scanner sc = new Scanner(System.in);
7         int n = sc.nextInt();
8         HashMap<String,Integer> hm = new HashMap<String,Integer>();
9         for(int i=0;i<n;i++)
10        {
11            hm.put(sc.next(),sc.nextInt());
12        }
13        System.out.println(hm);
14        String s = sc.next();
15        hm.remove(s);
16        System.out.print(hm);
17    }
18 }
```


| | Input | Expected | Got | |
|---|--|---------------------------------|---------------------------------|---|
| ✓ | 2 Alice 8 Mary 12 Alice | {Alice=8, Mary=12} {Mary=12} | {Alice=8, Mary=12} {Mary=12} | ✓ |

Passed all tests! ✓