LAB 06

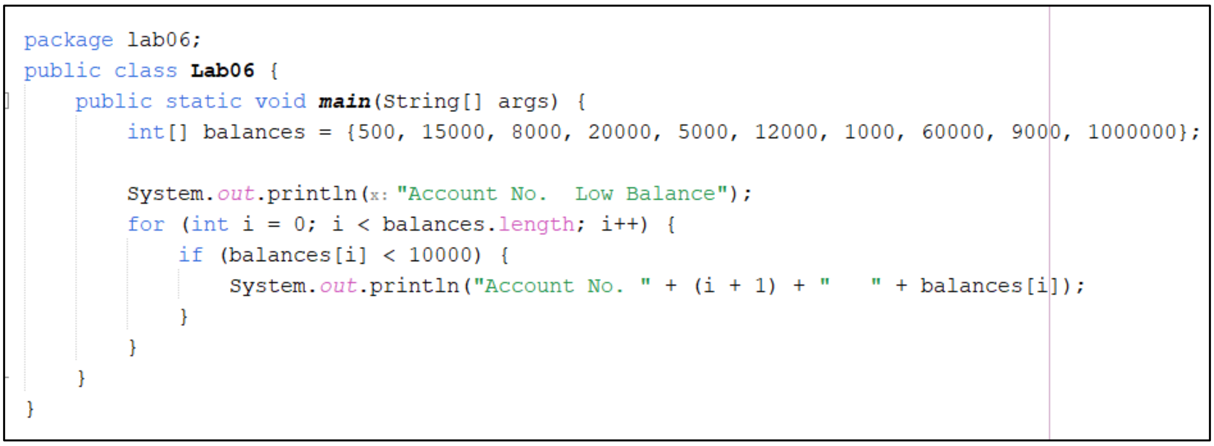
SEARCHING IN A LINEAR ARRAY

OBJECTIVE: To find an element in linear array using Linear Search and Binary Search.

LAB TASKS

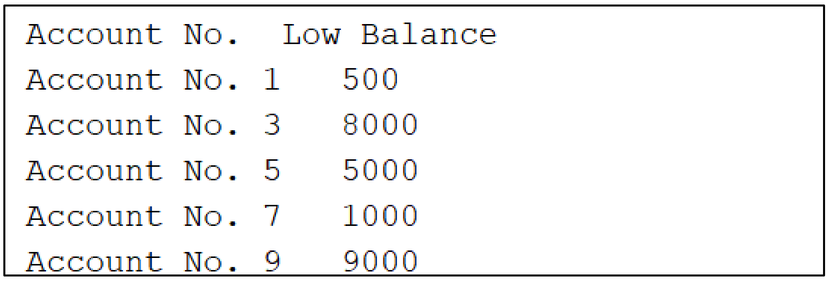
1. **Declare an array of size 10 to store account balances. Initialize with values 0 to 1000000. Check all arrays if any value is less than 10000. Show message: Account No. Low Balance**
 Account No. Low Balance

INPUT:



```
package lab06;
public class Lab06 {
    public static void main(String[] args) {
        int[] balances = {500, 15000, 8000, 20000, 5000, 12000, 1000, 60000, 9000, 1000000};

        System.out.println(x: "Account No.  Low Balance");
        for (int i = 0; i < balances.length; i++) {
            if (balances[i] < 10000) {
                System.out.println("Account No. " + (i + 1) + "    " + balances[i]);
            }
        }
    }
}
```

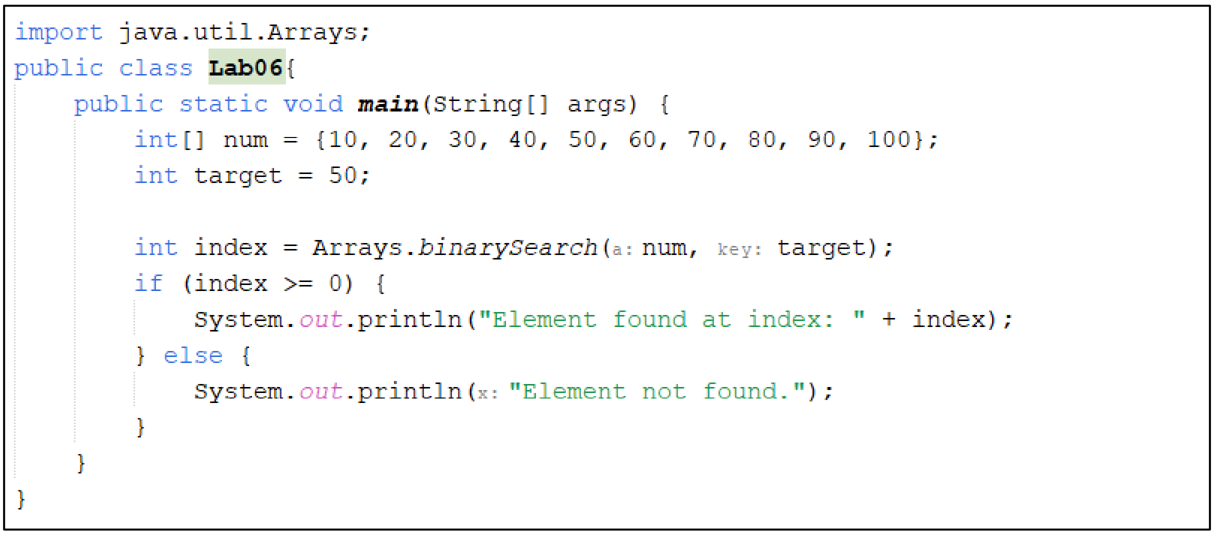
OUTPUT:



```
Account No.  Low Balance
Account No. 1    500
Account No. 3    8000
Account No. 5    5000
Account No. 7    1000
Account No. 9    9000
```

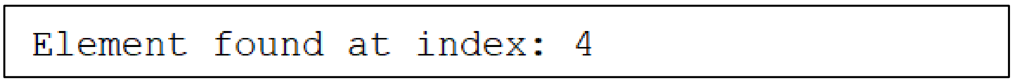
2. **Write a program to search in array using Array built-in class**

INPUT:



```
import java.util.Arrays;
public class Lab06{
    public static void main(String[] args) {
        int[] num = {10, 20, 30, 40, 50, 60, 70, 80, 90, 100};
        int target = 50;

        int index = Arrays.binarySearch(a: num, key: target);
        if (index >= 0) {
            System.out.println("Element found at index: " + index);
        } else {
            System.out.println(x: "Element not found.");
        }
    }
}
```

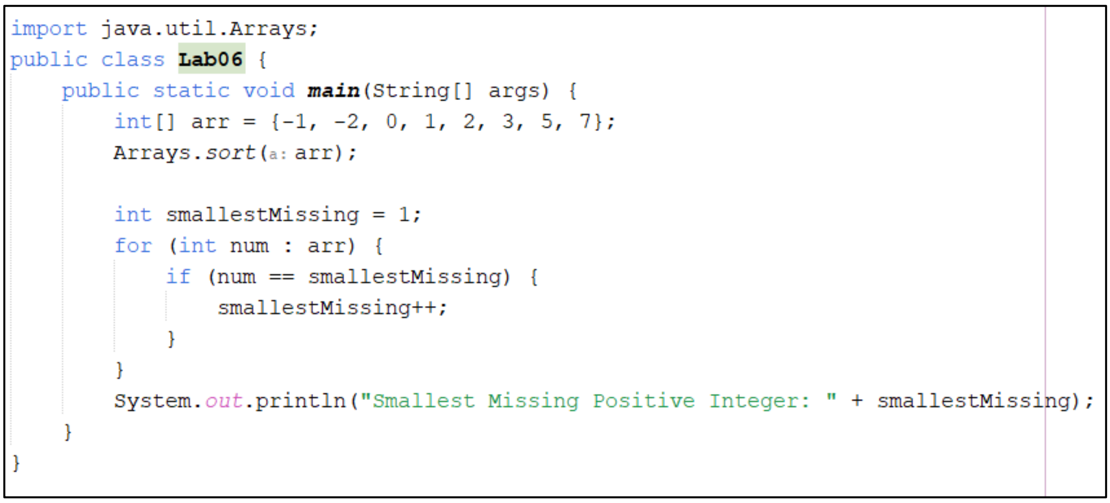
OUTPUT:



```
Element found at index: 4
```

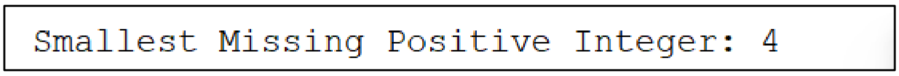
3. **Given an unsorted array arr of integers, find the smallest positive integer that is missing from the array. You need to implement this using binary search. The array can contain both negative numbers and positive numbers, and you can assume that the array does not have duplicates**

INPUT:



```
import java.util.Arrays;
public class Lab06 {
    public static void main(String[] args) {
        int[] arr = {-1, -2, 0, 1, 2, 3, 5, 7};
        Arrays.sort(a: arr);

        int smallestMissing = 1;
        for (int num : arr) {
            if (num == smallestMissing) {
                smallestMissing++;
            }
        }
        System.out.println("Smallest Missing Positive Integer: " + smallestMissing);
    }
}
```

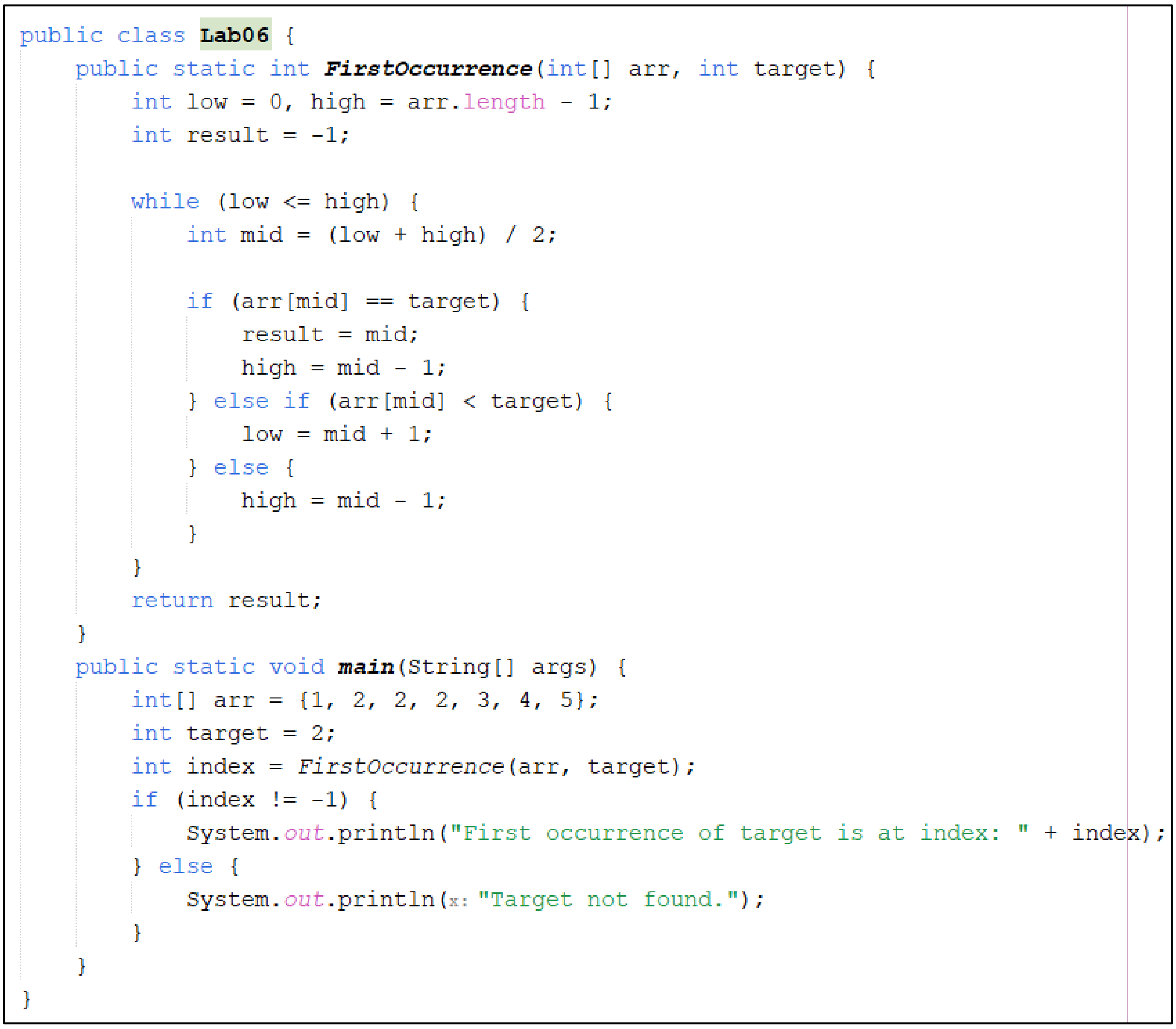
OUTPUT:



```
Smallest Missing Positive Integer: 4
```

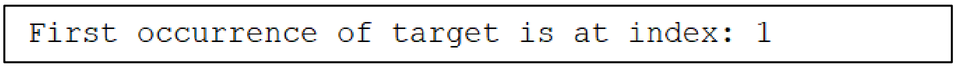
4. **You are given a sorted array arr[] and a target element target. Your task is to find the first occurrence of the target in the array using binary search. If the target is not found, return -1. You are given a sorted array arr[] and a target element target. Your task is to find the first occurrence of the target in the array using binary search. If the target is not found, return -1**

INPUT:



```java
public class Lab06 {
    public static int FirstOccurrence(int[] arr, int target) {
        int low = 0, high = arr.length - 1;
        int result = -1;

        while (low <= high) {
            int mid = (low + high) / 2;

            if (arr[mid] == target) {
                result = mid;
                high = mid - 1;
            } else if (arr[mid] < target) {
                low = mid + 1;
            } else {
                high = mid - 1;
            }
        }
        return result;
    }
    public static void main(String[] args) {
        int[] arr = {1, 2, 2, 2, 3, 4, 5};
        int target = 2;
        int index = FirstOccurrence(arr, target);
        if (index != -1) {
            System.out.println("First occurrence of target is at index: " + index);
        } else {
            System.out.println(x: "Target not found.");
        }
    }
}
```

OUTPUT:

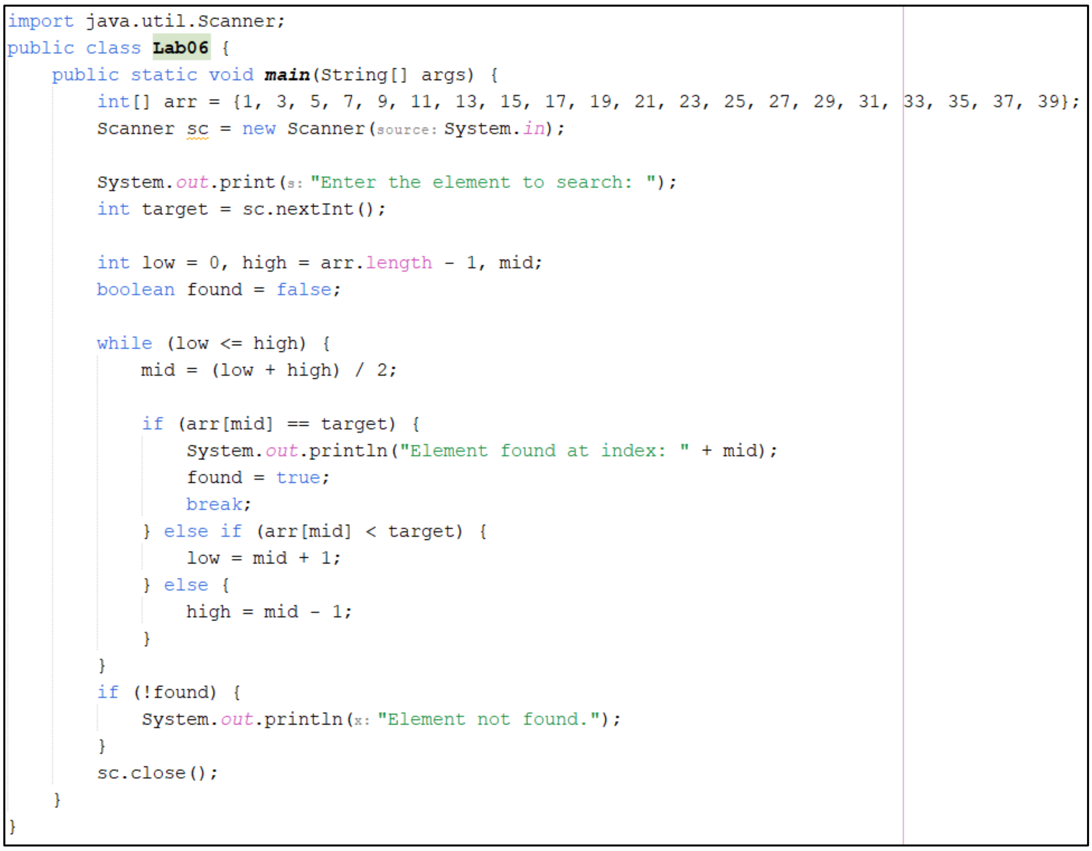


```
First occurrence of target is at index: 1
```

HOME TASK

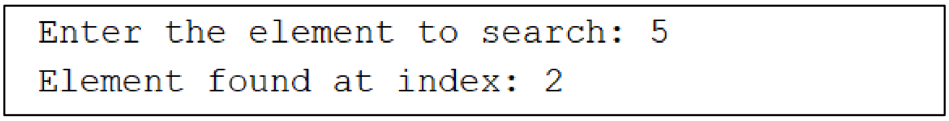
1. **Write a program initializing array of size 20 and search an element using binary search**

INPUT:

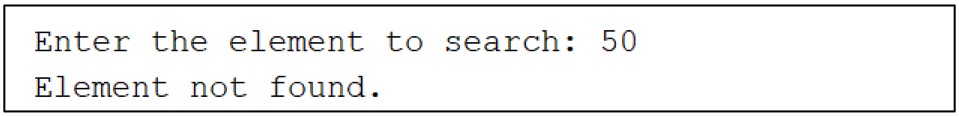


```java
import java.util.Scanner;
public class Lab06 {
    public static void main(String[] args) {
        int[] arr = {1, 3, 5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 27, 29, 31, 33, 35, 37, 39};
        Scanner sc = new Scanner(source: System.in);

        System.out.print(s: "Enter the element to search: ");
        int target = sc.nextInt();

        int low = 0, high = arr.length - 1, mid;
        boolean found = false;

        while (low <= high) {
            mid = (low + high) / 2;

            if (arr[mid] == target) {
                System.out.println("Element found at index: " + mid);
                found = true;
                break;
            } else if (arr[mid] < target) {
                low = mid + 1;
            } else {
                high = mid - 1;
            }
        }
        if (!found) {
            System.out.println(x: "Element not found.");
        }
        sc.close();
    }
}
```

OUTPUT:



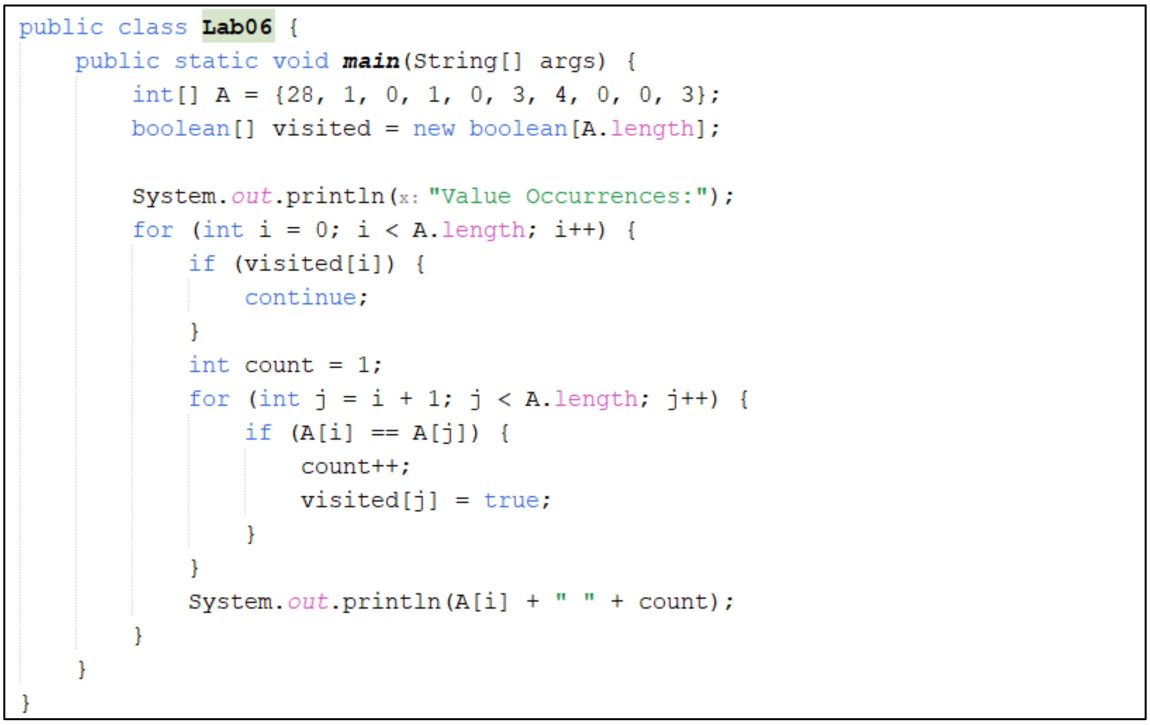
```
Enter the element to search: 5
Element found at index: 2
```



```
Enter the element to search: 50
Element not found.
```

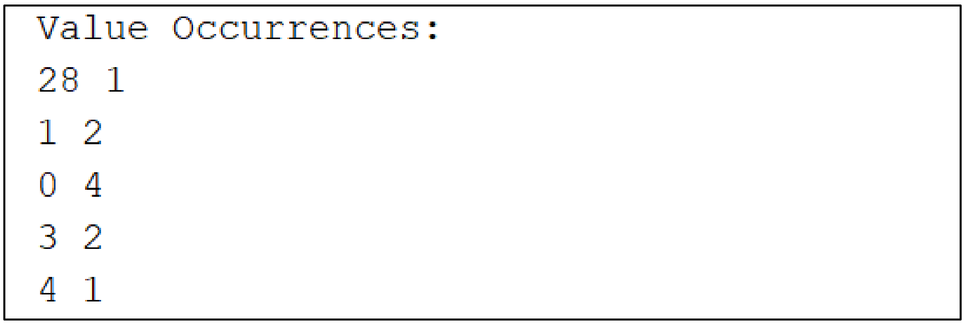
2. **Write a function called occurrences that, given an array of numbers A, prints all the distinct values in A each followed by its number of occurrences**

INPUT:



```
public class Lab06 {
    public static void main(String[] args) {
        int[] A = {28, 1, 0, 1, 0, 3, 4, 0, 0, 3};
        boolean[] visited = new boolean[A.length];

        System.out.println(x: "Value Occurrences:");
        for (int i = 0; i < A.length; i++) {
            if (visited[i]) {
                continue;
            }
            int count = 1;
            for (int j = i + 1; j < A.length; j++) {
                if (A[i] == A[j]) {
                    count++;
                    visited[j] = true;
                }
            }
            System.out.println(A[i] + " " + count);
        }
    }
}
```

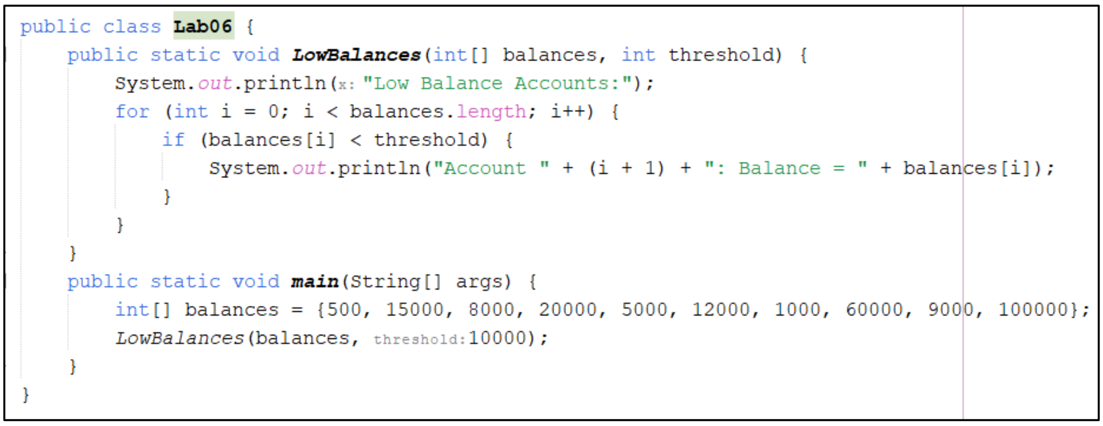
OUTPUT:



```
Value Occurrences:
28 1
1 2
0 4
3 2
4 1
```

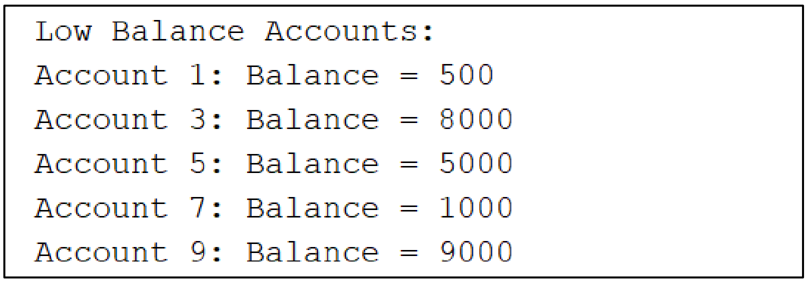
3. **Assume a bank's system needs to identify accounts with critically low balances and alert the user. Test the function with various balance values to ensure it correctly identifies all accounts below the threshold**

INPUT:



```java
public class Lab06 {
    public static void LowBalances(int[] balances, int threshold) {
        System.out.println(x: "Low Balance Accounts:");
        for (int i = 0; i < balances.length; i++) {
            if (balances[i] < threshold) {
                System.out.println("Account " + (i + 1) + ": Balance = " + balances[i]);
            }
        }
    }
    public static void main(String[] args) {
        int[] balances = {500, 15000, 8000, 20000, 5000, 12000, 1000, 60000, 9000, 100000};
        LowBalances(balances, threshold:10000);
    }
}
```

OUTPUT:



```
Low Balance Accounts:
Account 1: Balance = 500
Account 3: Balance = 8000
Account 5: Balance = 5000
Account 7: Balance = 1000
Account 9: Balance = 9000
```