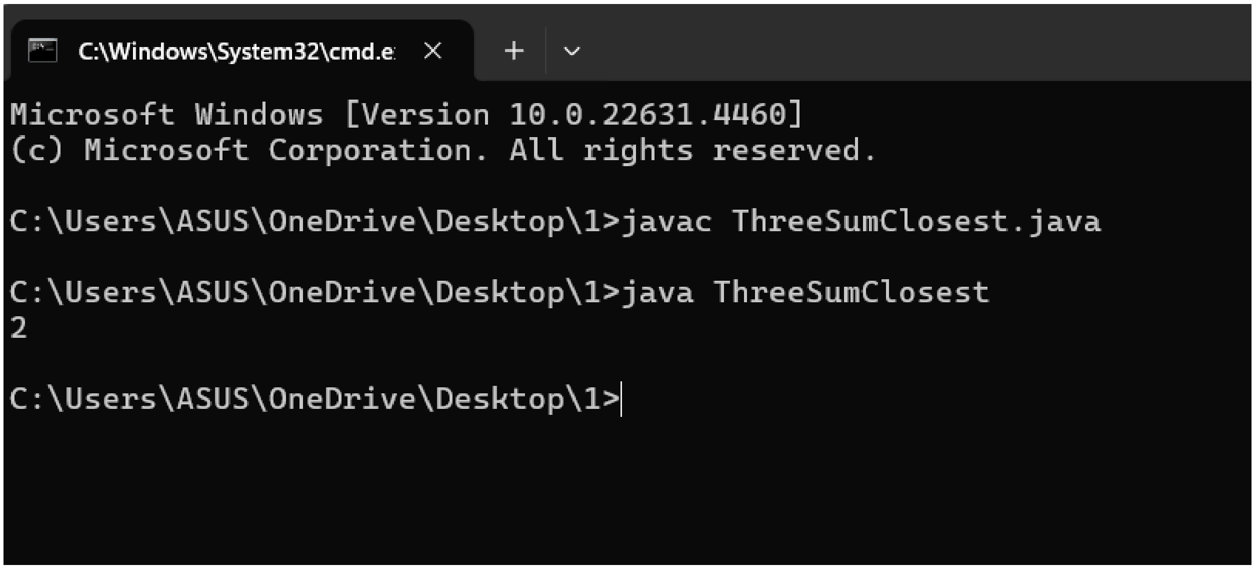
1)3Sum Closest

```
import java.util.Arrays;

public class ThreeSumClosest {
    public static void main(String[] args) {
        int[] nums = {-1, 2, 1, -4};
        int target = 1;

        System.out.println(threeSumClosest(nums, target));
    }

    public static int threeSumClosest(int[] nums, int target) {
        Arrays.sort(nums);
        int closestSum = Integer.MAX_VALUE;

        for (int i = 0; i < nums.length - 2; i++) {
            int left = i + 1;
            int right = nums.length - 1;

            while (left < right) {
                int sum = nums[i] + nums[left] + nums[right];

                if (sum == target) {
                    return sum;
                }

                if (Math.abs(sum - target) < Math.abs(closestSum - target)) {
                    closestSum = sum;
                }

                if (sum < target) {
                    left++;
                } else {
                    right--;
                }
            }
        }

        return closestSum;
    }
}
```

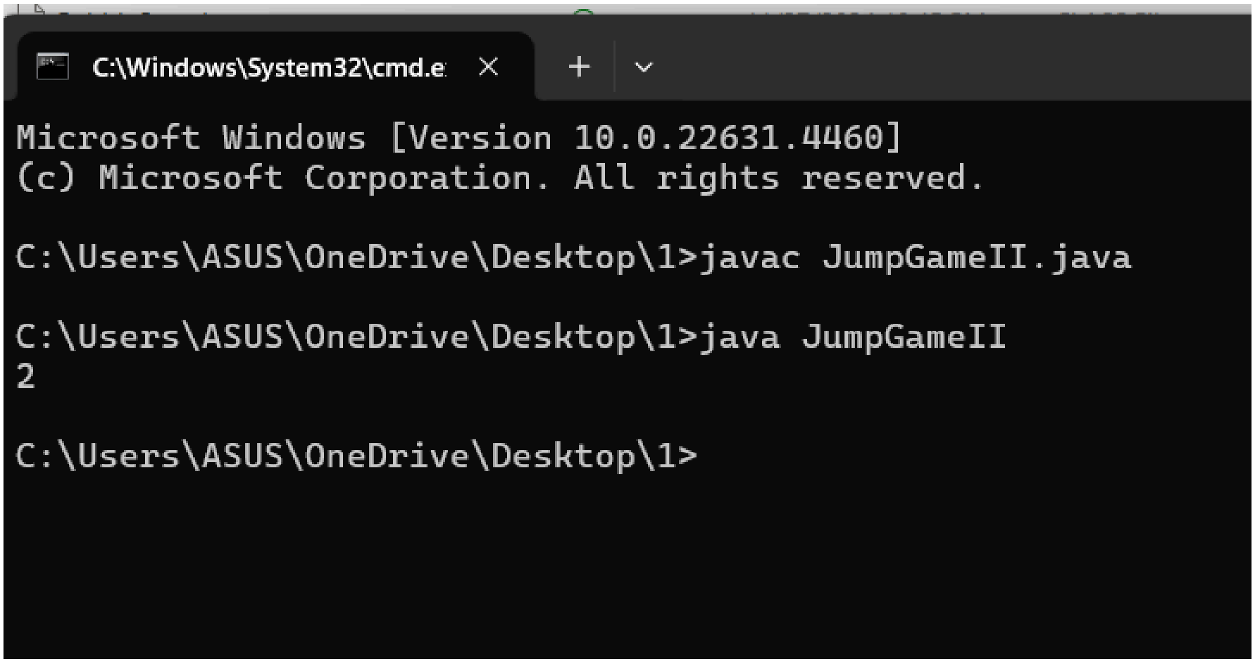


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac ThreeSumClosest.java

C:\Users\ASUS\OneDrive\Desktop\1>java ThreeSumClosest
2

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(n^2)

2)Jump Game II

```
public class JumpGameII {
    public static void main(String[] args) {
        int[] nums = {2, 3, 1, 1, 4};

        System.out.println(jump(nums));     }

    public static int jump(int[] nums) {
        if (nums.length <= 1) {
            return 0;
        }
        int jumps = 0;
        int currentEnd = 0;
        int farthest = 0;

        for (int i = 0; i < nums.length - 1; i++) {
            farthest = Math.max(farthest, i + nums[i]);

            if (i == currentEnd) {
                jumps++;
                currentEnd = farthest;

                if (currentEnd >= nums.length - 1) {
                    break;
                }
            }
        }

        return jumps;
    }
}
```

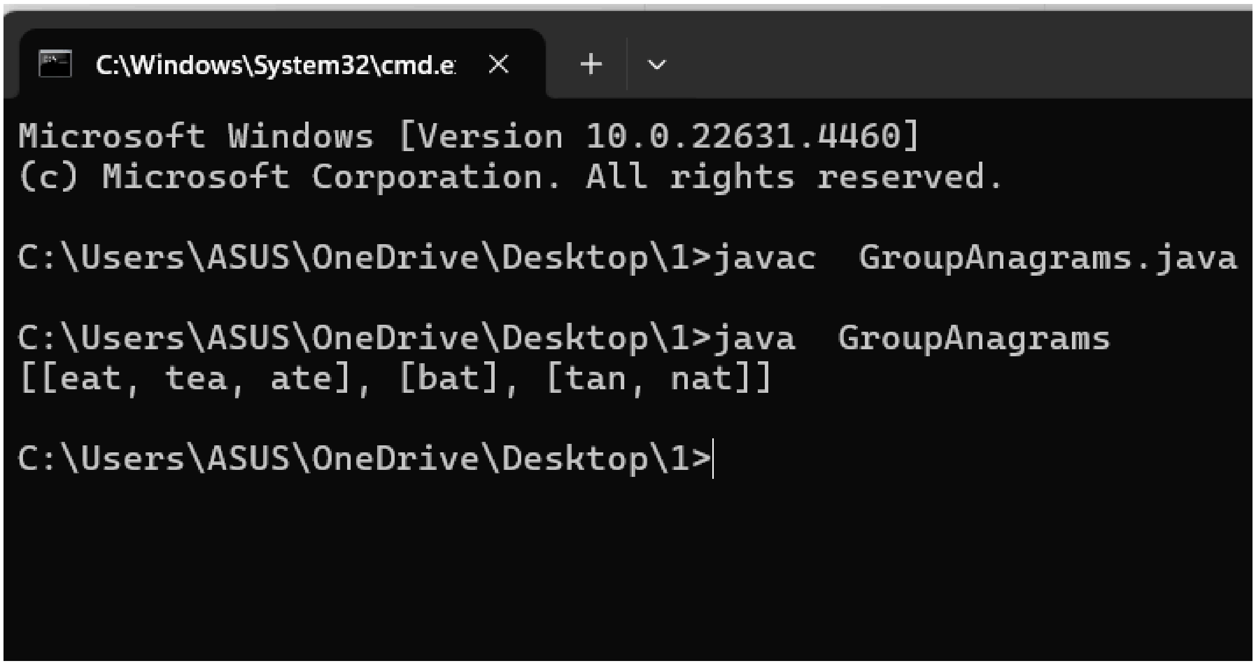


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac JumpGameII.java

C:\Users\ASUS\OneDrive\Desktop\1>java JumpGameII
2

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(n)

3)Group Anagrams

```
import java.util.*;

public class GroupAnagrams {
    public static void main(String[] args) {
        String[] strs = {"eat", "tea", "tan", "ate", "nat", "bat"};
        System.out.println(groupAnagrams(strs));
    }

    public static List<List<String>> groupAnagrams(String[] strs) {
        Map<String, List<String>> map = new HashMap<>();

        for (String str : strs) {
            char[] chars = str.toCharArray();
            Arrays.sort(chars);
            String sortedStr = new String(chars);

            map.putIfAbsent(sortedStr, new ArrayList<>());
            map.get(sortedStr).add(str);
        }

        return new ArrayList<>(map.values());
    }
}
```

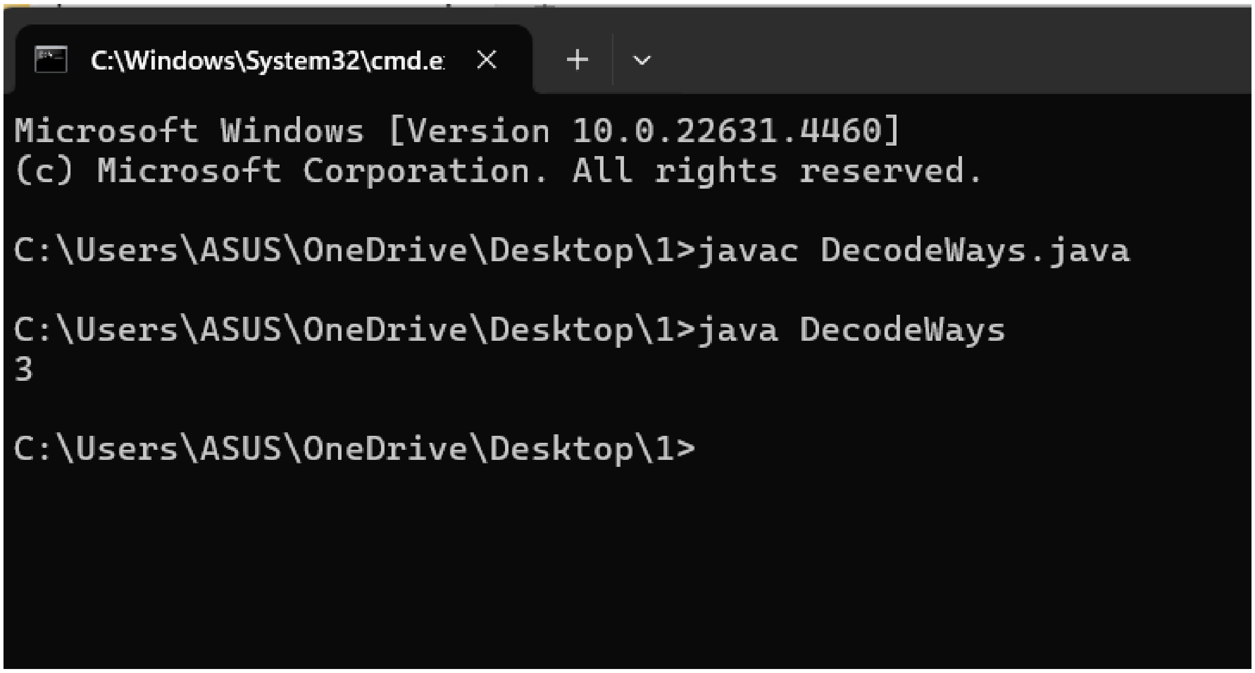


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac  GroupAnagrams.java

C:\Users\ASUS\OneDrive\Desktop\1>java  GroupAnagrams
[[eat, tea, ate], [bat], [tan, nat]]

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(nk logk)

4)Decode Ways

```
public class DecodeWays {
    public static void main(String[] args) {
        String s = "226";
        System.out.println(numDecodings(s));
    }

    public static int numDecodings(String s) {
        if (s == null || s.length() == 0 || s.charAt(0) == '0') {
            return 0;
        }
        int n = s.length();
        int[] dp = new int[n + 1];
        dp[0] = 1;
        dp[1] = 1;

        for (int i = 2; i <= n; i++) {
            if (s.charAt(i - 1) != '0') {
                dp[i] += dp[i - 1];
            }

            int twoDigit = Integer.parseInt(s.substring(i - 2, i));
            if (twoDigit >= 10 && twoDigit <= 26) {
                dp[i] += dp[i - 2];
            }
        }

        return dp[n];
    }
}
```

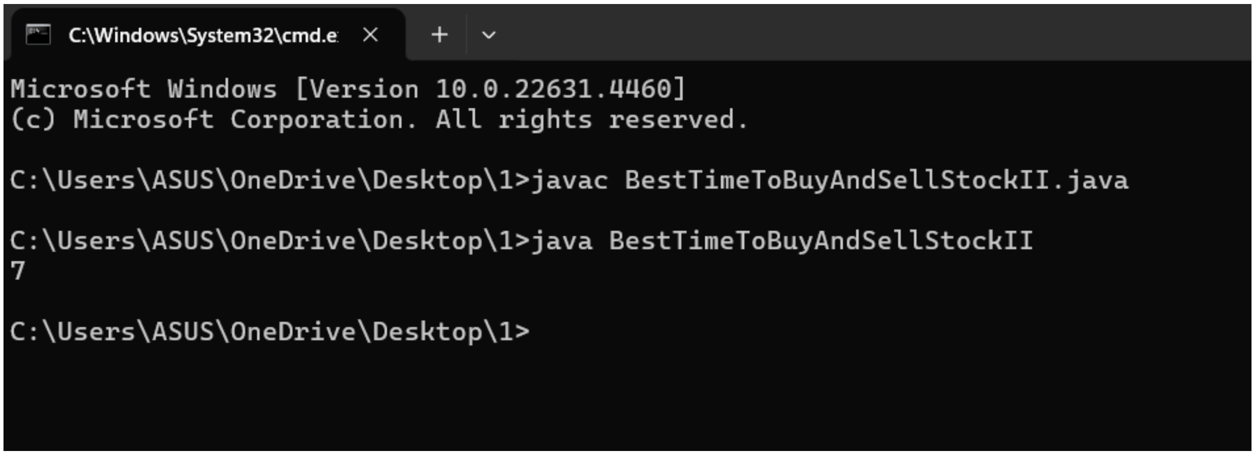


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac DecodeWays.java

C:\Users\ASUS\OneDrive\Desktop\1>java DecodeWays
3

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(n)

5)Best Time to Buy and Sell Stock II

```
public class BestTimeToBuyAndSellStockII {
    public static void main(String[] args) {
        int[] prices = {7, 1, 5, 3, 6, 4};
        System.out.println(maxProfit(prices));
    }

    public static int maxProfit(int[] prices) {
        int profit = 0;

        for (int i = 1; i < prices.length; i++) {
            if (prices[i] > prices[i - 1]) {
                profit += prices[i] - prices[i - 1];
            }
        }

        return profit;
    }
}
```

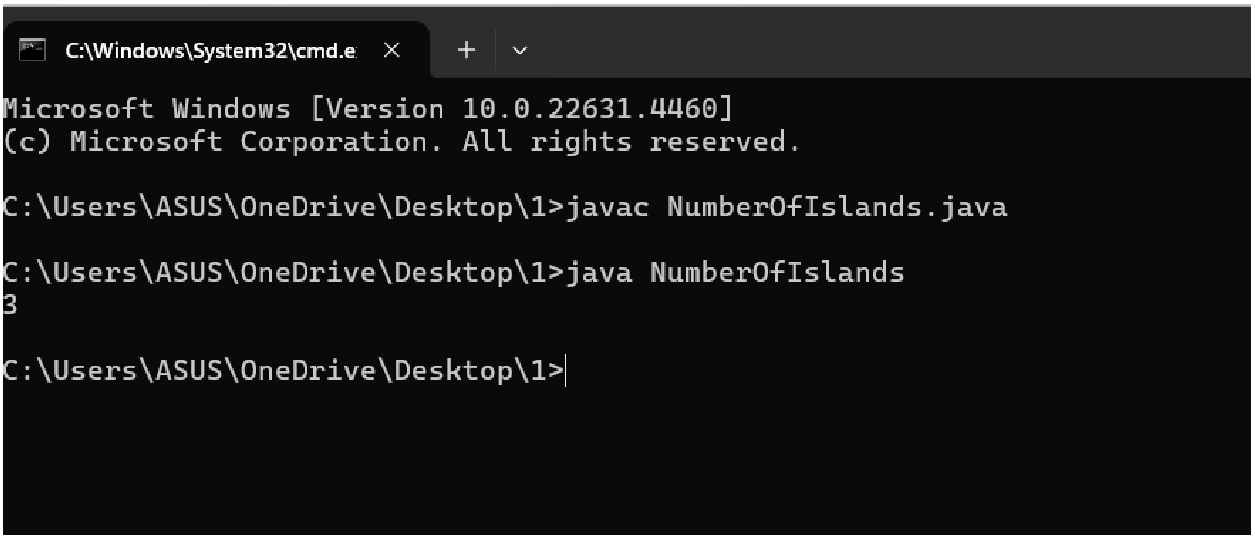


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac BestTimeToBuyAndSellStockII.java

C:\Users\ASUS\OneDrive\Desktop\1>java BestTimeToBuyAndSellStockII
7

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(n)

6)Number Of Islands

```
public class NumberOfIslands {
    public static void main(String[] args) {
        char[][] grid = {
            {'1', '1', '0', '0', '0'},
            {'1', '1', '0', '0', '0'},
            {'0', '0', '1', '0', '0'},
            {'0', '0', '0', '1', '1'}
        };
        System.out.println(numIslands(grid));      }

    public static int numIslands(char[][] grid) {
        if (grid == null || grid.length == 0) {
            return 0;
        }

        int count = 0;
        int rows = grid.length;
        int cols = grid[0].length;

        for (int i = 0; i < rows; i++) {
            for (int j = 0; j < cols; j++) {
                if (grid[i][j] == '1') {
                    count++;
                    dfs(grid, i, j);
                }
            }
        }

        return count;
    }
    private static void dfs(char[][] grid, int i, int j) {
        if (i < 0 || i >= grid.length || j < 0 || j >= grid[0].length || grid[i][j] == '0') {
            return;
        }

        grid[i][j] = '0';

        dfs(grid, i - 1, j);
        dfs(grid, i + 1, j);
        dfs(grid, i, j - 1);
        dfs(grid, i, j + 1);
    }
}
```



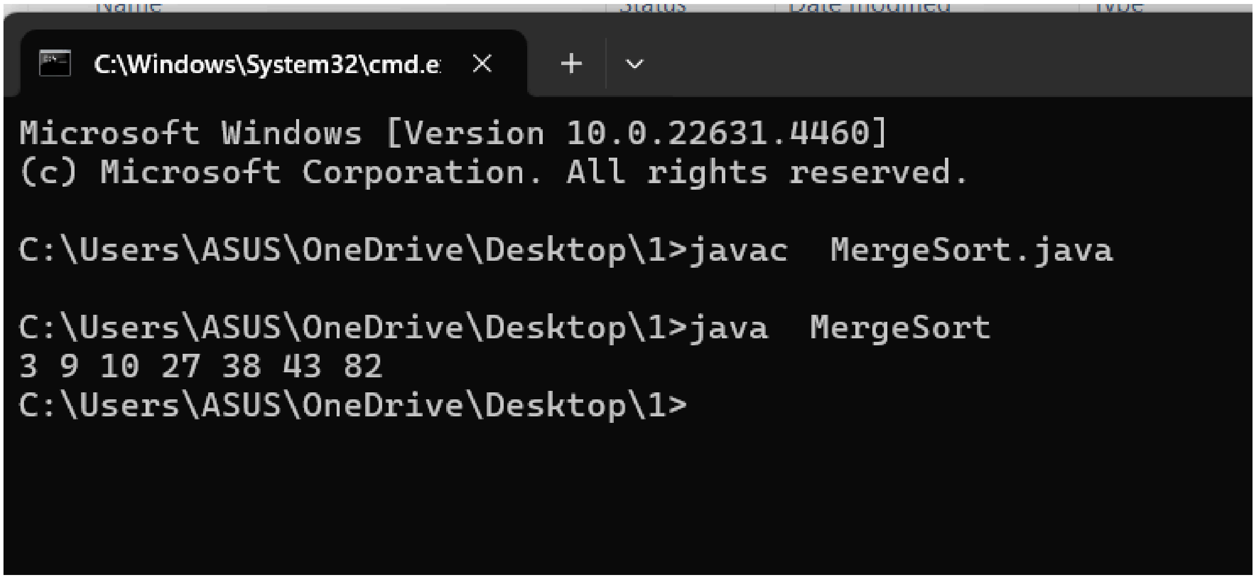
```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac NumberOfIslands.java

C:\Users\ASUS\OneDrive\Desktop\1>java NumberOfIslands
3

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(mxn)

7)Merge sort

```
public class MergeSort {
    public static void main(String[] args) {
        int[] arr = {38, 27, 43, 3, 9, 82, 10};
        mergeSort(arr, 0, arr.length - 1);
        for (int num : arr) {
            System.out.print(num + " ");
        }
    }

    public static void mergeSort(int[] arr, int left, int right) {
        if (left < right) {
            int mid = left + (right - left) / 2;
            mergeSort(arr, left, mid);
            mergeSort(arr, mid + 1, right);
            merge(arr, left, mid, right);
        }
    }

    public static void merge(int[] arr, int left, int mid, int right) {
        int n1 = mid - left + 1;
        int n2 = right - mid;

        int[] leftArray = new int[n1];
        int[] rightArray = new int[n2];

        for (int i = 0; i < n1; i++) {
            leftArray[i] = arr[left + i];
        }
        for (int i = 0; i < n2; i++) {
            rightArray[i] = arr[mid + 1 + i];
        }

        int i = 0, j = 0, k = left;

        while (i < n1 && j < n2) {
            if (leftArray[i] <= rightArray[j]) {
                arr[k++] = leftArray[i++];
            } else {
                arr[k++] = rightArray[j++];
            }
        }
```

```
        while (i < n1) {
            arr[k++] = leftArray[i++];
        }

        while (j < n2) {
            arr[k++] = rightArray[j++];
        }
    }
}
```

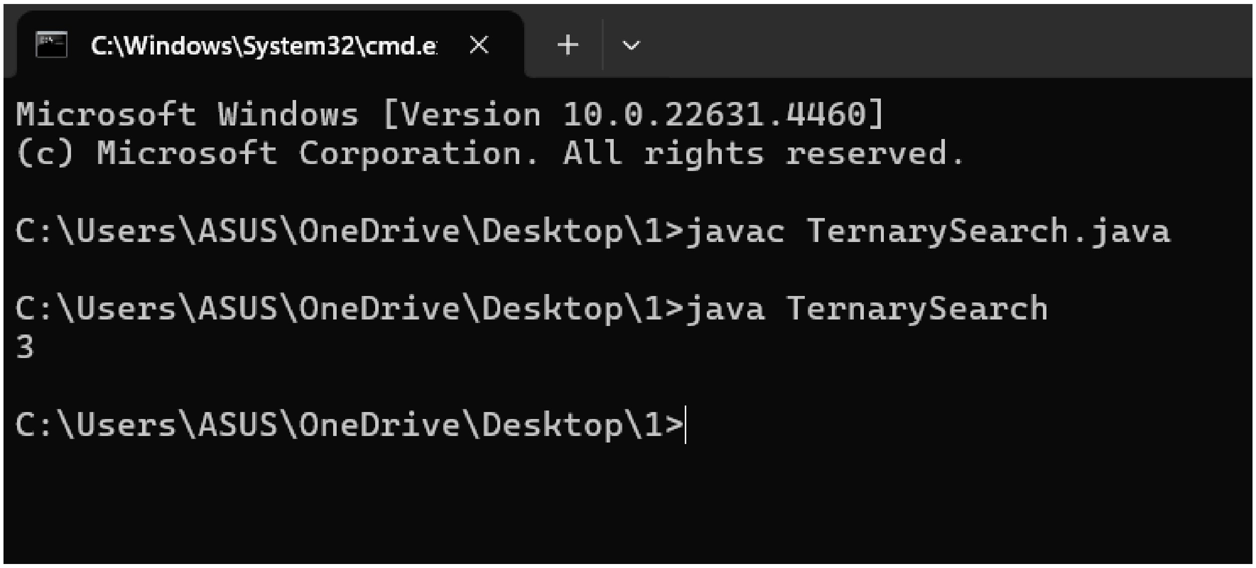


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac  MergeSort.java

C:\Users\ASUS\OneDrive\Desktop\1>java  MergeSort
3 9 10 27 38 43 82
C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(n logn)

8)Ternary Search

```
public class TernarySearch {
    public static void main(String[] args) {
        int[] arr = {1, 3, 5, 7, 9, 11, 13};
        int target = 7;
        System.out.println(ternarySearch(arr, 0, arr.length - 1, target));
    }
    public static int ternarySearch(int[] arr, int left, int right, int target) {
        if (right >= left) {
            int mid1 = left + (right - left) / 3;
            int mid2 = right - (right - left) / 3;

            if (arr[mid1] == target) {
                return mid1;
            }
            if (arr[mid2] == target) {
                return mid2;
            }

            if (target < arr[mid1]) {
                return ternarySearch(arr, left, mid1 - 1, target);
            } else if (target > arr[mid2]) {
                return ternarySearch(arr, mid2 + 1, right, target);
            } else {
                return ternarySearch(arr, mid1 + 1, mid2 - 1, target);
            }
        }

        return -1;
    }
}
```

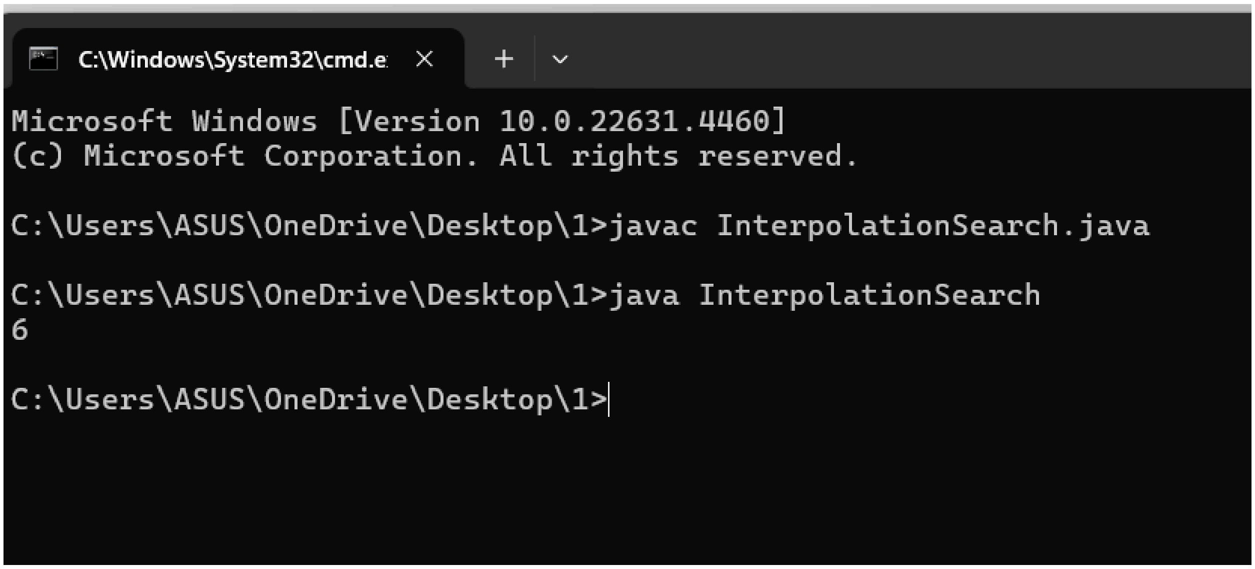


```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac TernarySearch.java

C:\Users\ASUS\OneDrive\Desktop\1>java TernarySearch
3

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(log3n)

9)Interpolation Search

```
public class InterpolationSearch {
    public static void main(String[] args) {
        int[] arr = {10, 20, 30, 40, 50, 60, 70, 80, 90};
        int target = 70;
        System.out.println(interpolationSearch(arr, target));
    }

    public static int interpolationSearch(int[] arr, int target) {
        int low = 0, high = arr.length - 1;

        while (low <= high && target >= arr[low] && target <= arr[high]) {
            int pos = low + (target - arr[low]) * (high - low) / (arr[high] - arr[low]);

            if (arr[pos] == target) {
                return pos;
            } else if (arr[pos] < target) {
                low = pos + 1;
            } else {
                high = pos - 1;
            }
        }

        return -1;
    }
}
```



```
Microsoft Windows [Version 10.0.22631.4460]
(c) Microsoft Corporation. All rights reserved.

C:\Users\ASUS\OneDrive\Desktop\1>javac InterpolationSearch.java

C:\Users\ASUS\OneDrive\Desktop\1>java InterpolationSearch
6

C:\Users\ASUS\OneDrive\Desktop\1>
```

Time:O(1)