VIJAY M(IT)

22IT125

CODING PRACTICES AND PROBLEMS

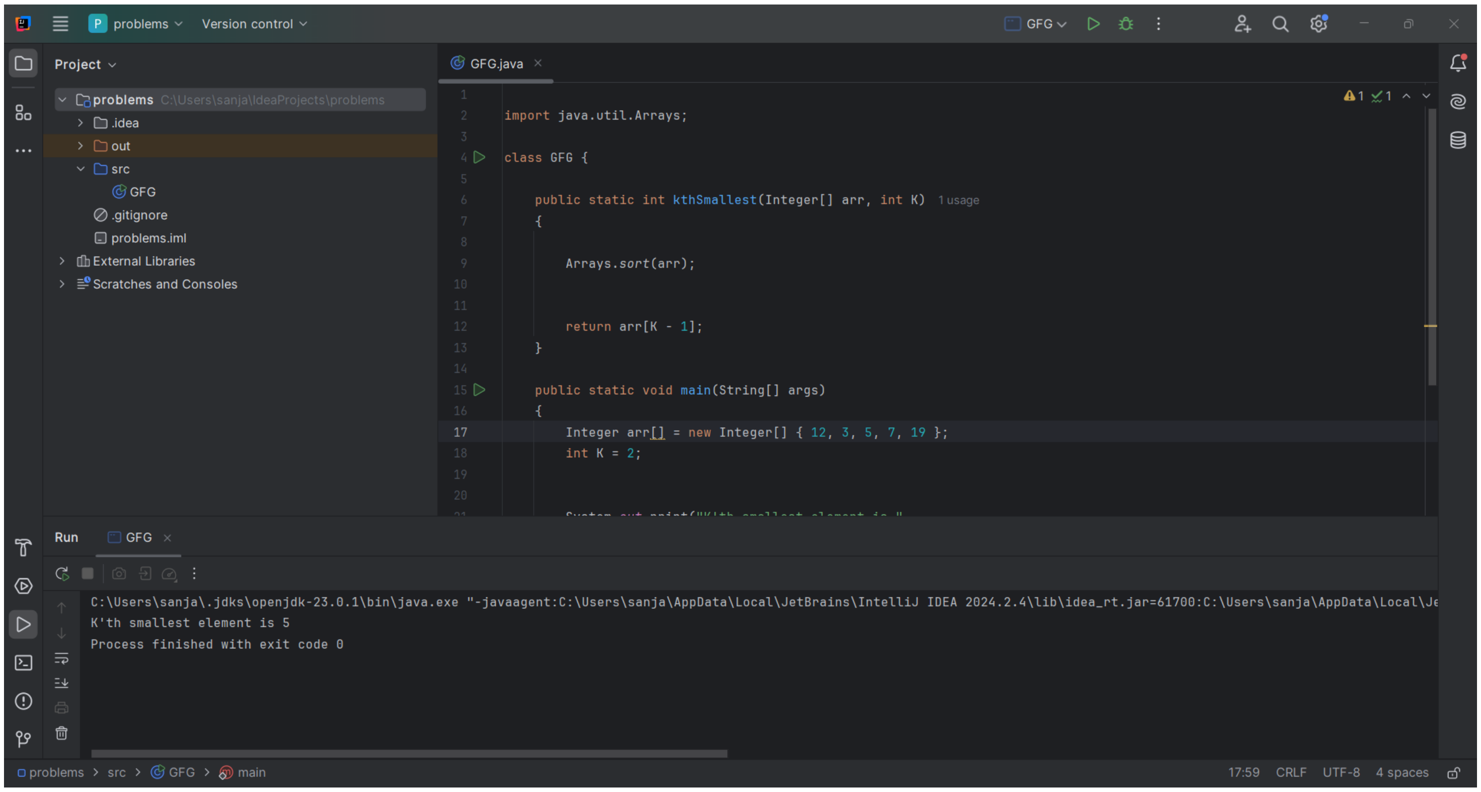
1.Kth Smallest Elements

Time Complexity: O(n log(n))

SOLUTION:

```
import java.util.Arrays;

class GFG {

   public static int kthSmallest(Integer[] arr, int K)
   {

      Arrays.sort(arr);


      return arr[K - 1];
   }

   public static void main(String[] args)
   {
      Integer arr[] = new Integer[] { 12, 3, 5, 7, 19 };
      int K = 2;


      System.out.print("K'th smallest element is "
            + kthSmallest(arr, K));
   }
}
```

OUTPUT:



```java

import java.util.Arrays;

class GFG {

    public static int kthSmallest(Integer[] arr, int K)
    {

        Arrays.sort(arr);


        return arr[K - 1];
    }

    public static void main(String[] args)
    {
        Integer arr[] = new Integer[] { 12, 3, 5, 7, 19 };
        int K = 2;


```

```
C:\Users\sanja\.jdks\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\sanja\AppData\Local\JetBrains\IntelliJ IDEA 2024.2.4\lib\idea_rt.jar=61700:C:\Users\sanja\AppData\Local\Je
K'th smallest element is 5
Process finished with exit code 0
```

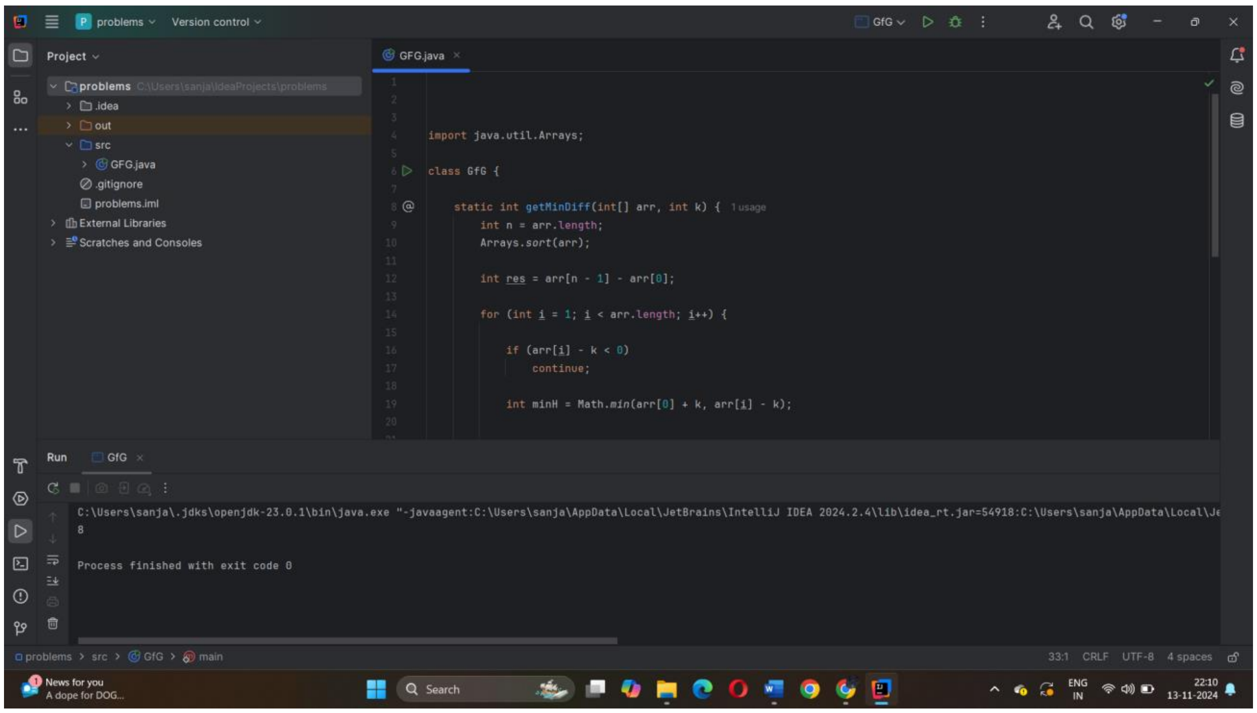
2.Minimize Height II

Time Complexity: O(nlogn)

SOLUTION:

```java
import java.util.Arrays;

class GfG {

  static int getMinDiff(int[] arr, int k) {
     int n = arr.length;
     Arrays.sort(arr);

     int res = arr[n - 1] - arr[0];

     for (int i = 1; i < arr.length; i++) {
         if (arr[i] - k < 0)
             continue;
         int minH = Math.min(arr[0] + k, arr[i] - k);

         int maxH = Math.max(arr[i - 1] + k, arr[n - 1] - k);

         res = Math.min(res, maxH - minH);
     }
     return res;
  }

  public static void main(String[] args) {
     int k = 6;
     int[] arr = {12, 6, 4, 15, 17, 10};

     int ans = getMinDiff(arr, k);
     System.out.println(ans);
  }
}
```

OUTPUT:



```
import java.util.Arrays;

class GfG {

    static int getMinDiff(int[] arr, int k) {
        int n = arr.length;
        Arrays.sort(arr);

        int res = arr[n - 1] - arr[0];

        for (int i = 1; i < arr.length; i++) {

            if (arr[i] - k < 0)
                continue;

            int minH = Math.min(arr[0] + k, arr[i] - k);
```

```
C:\Users\sanja\.jdks\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\sanja\AppData\Local\JetBrains\IntelliJ IDEA 2024.2.4\lib\idea_rt.jar=54918:C:\Users\sanja\AppData\Local\Je
8

Process finished with exit code 0
```

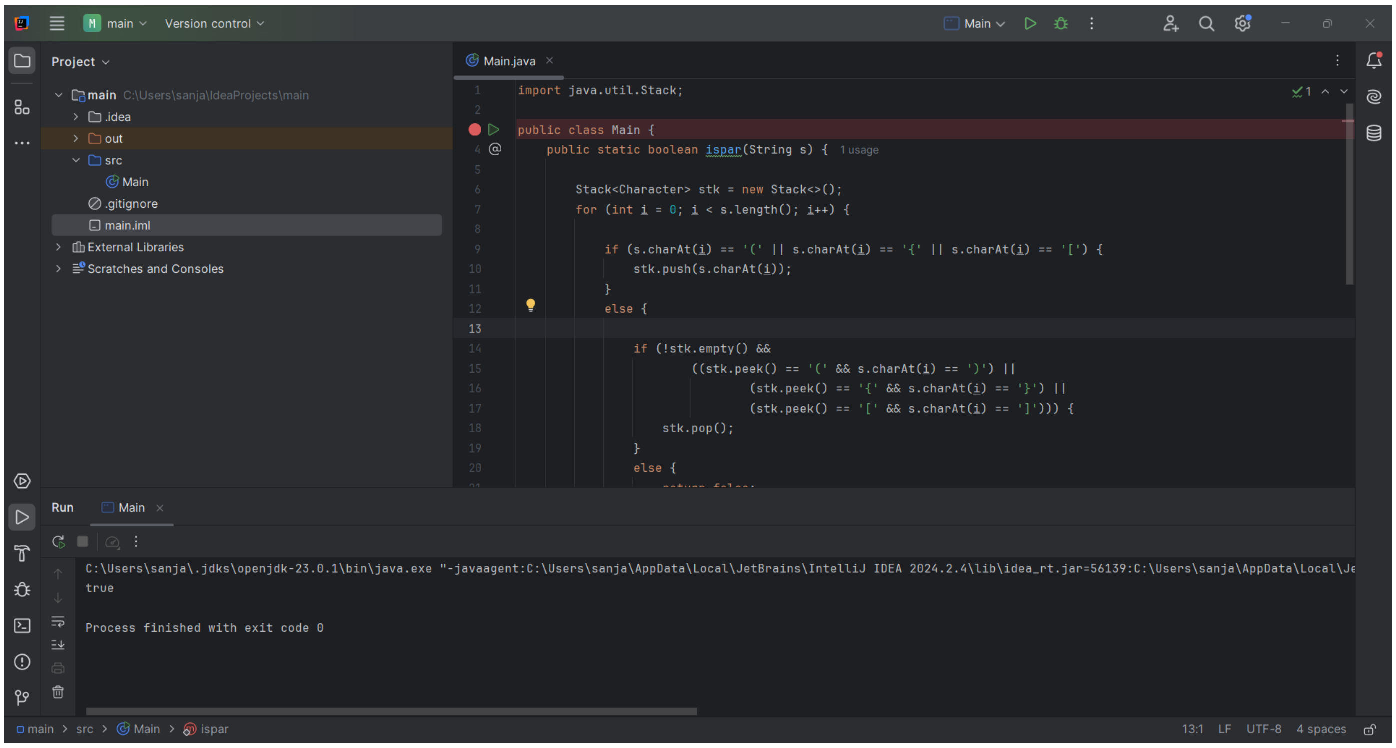
3.Paranthesis Checker

Time Complexity:O(n)

SOLUTION:

```
import java.util.Stack;
public class Main {
  public static boolean ispar(String s) {

      Stack<Character> stk = new Stack<>();
      for (int i = 0; i < s.length(); i++) {

          if (s.charAt(i) == '(' || s.charAt(i) == '{' || s.charAt(i) == '[') {
              stk.push(s.charAt(i));
          }
          else {

              if (!stk.empty() &&
                      ((stk.peek() == '(' && s.charAt(i) == ')') ||
                              (stk.peek() == '{' && s.charAt(i) == '}') ||
                              (stk.peek() == '[' && s.charAt(i) == ']'))) {
                  stk.pop();
              }
              else {
                  return false;
              }
          }
      }



      return stk.empty();
  }

  public static void main(String[] args) {
        String s = "{()}[]";
        if (ispar(s))
            System.out.println("true");
        else
            System.out.println("false");
    }
}
```

OUTPUT:



```java
import java.util.Stack;

public class Main {
    public static boolean ispar(String s) {  1 usage

        Stack<Character> stk = new Stack<>();
        for (int i = 0; i < s.length(); i++) {

            if (s.charAt(i) == '(' || s.charAt(i) == '{' || s.charAt(i) == '[') {
                stk.push(s.charAt(i));
            }
            else {

                if (!stk.empty() &&
                        ((stk.peek() == '(' && s.charAt(i) == ')') ||
                                (stk.peek() == '{' && s.charAt(i) == '}') ||
                                (stk.peek() == '[' && s.charAt(i) == ']'))) {
                    stk.pop();
                }
                else {
```

```
C:\Users\sanja\.jdks\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\sanja\AppData\Local\JetBrains\IntelliJ IDEA 2024.2.4\lib\idea_rt.jar=56139:C:\Users\sanja\AppData\Local\Je
true

Process finished with exit code 0
```

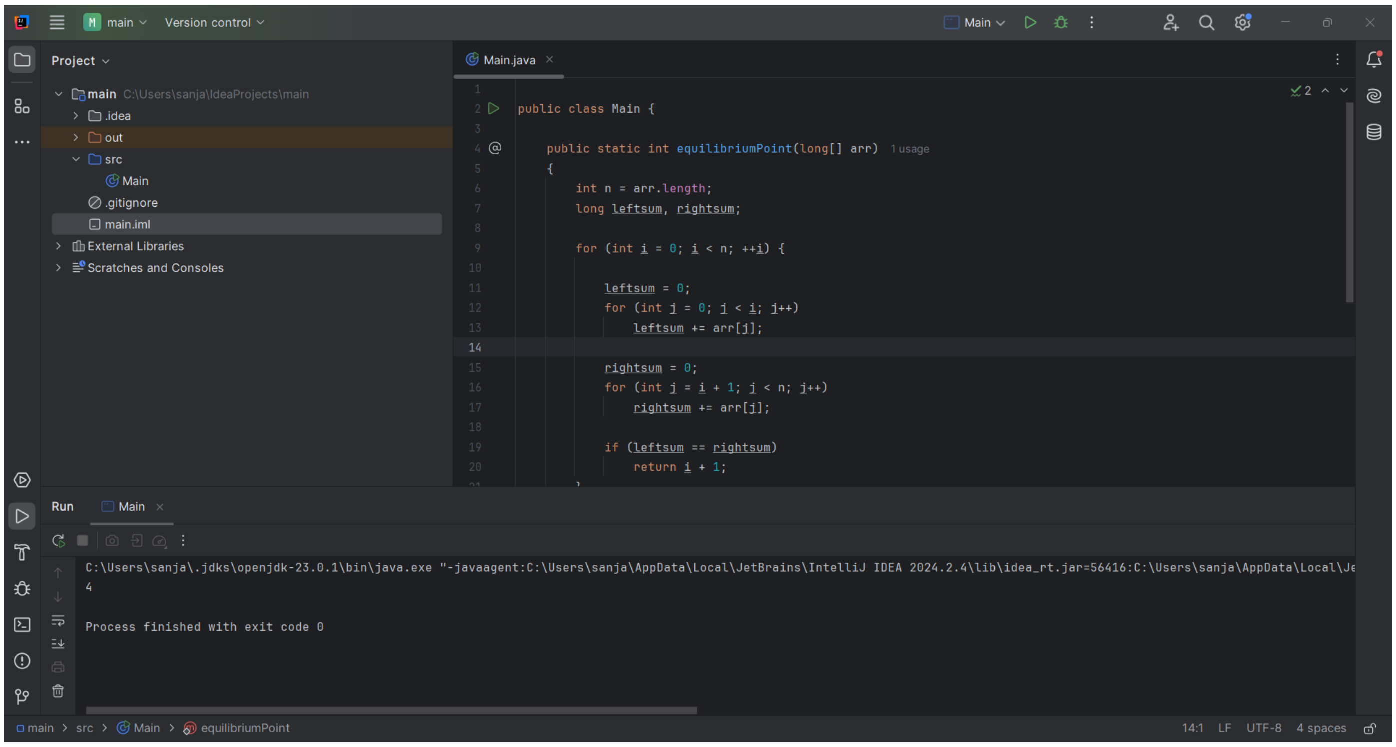
4. Equilibrium Point:

Time complexity:O(N^2)

SOLUTION:

```
public class Main {

   public static int equilibriumPoint(long[] arr)
   {
      int n = arr.length;
      long leftsum, rightsum;

      for (int i = 0; i < n; ++i) {

         leftsum = 0;
         for (int j = 0; j < i; j++)
            leftsum += arr[j];

         rightsum = 0;
         for (int j = i + 1; j < n; j++)
            rightsum += arr[j];

         if (leftsum == rightsum)
            return i + 1;
      }
      return -1;
   }

   public static void main(String[] args)
   {
      long[] arr = { -7, 1, 5, 2, -4, 3, 0 };

      System.out.println(equilibriumPoint(arr));
   }
}
```

OUTPUT:



```java

public class Main {

    public static int equilibriumPoint(long[] arr)
    {
        int n = arr.length;
        long leftsum, rightsum;

        for (int i = 0; i < n; ++i) {

            leftsum = 0;
            for (int j = 0; j < i; j++)
                leftsum += arr[j];

            rightsum = 0;
            for (int j = i + 1; j < n; j++)
                rightsum += arr[j];

            if (leftsum == rightsum)
                return i + 1;
```

```
C:\Users\sanja\.jdks\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\sanja\AppData\Local\JetBrains\IntelliJ IDEA 2024.2.4\lib\idea_rt.jar=56416:C:\Users\sanja\AppData\Local\Je
4

Process finished with exit code 0
```

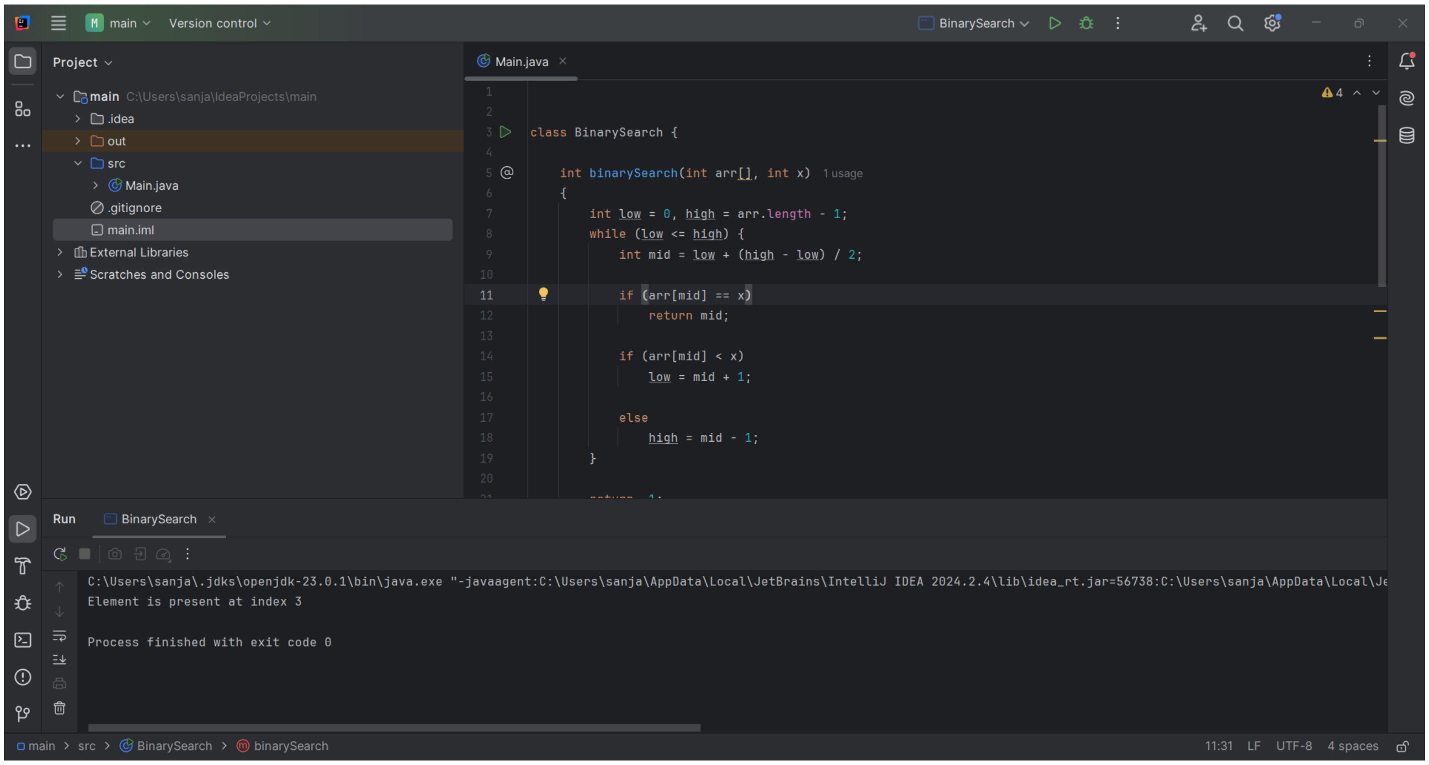
5.Binary Search:

Time Complexity:O(log N)

SOLUTION:

```java
class BinarySearch {

  int binarySearch(int arr[], int x)
  {
    int low = 0, high = arr.length - 1;
    while (low <= high) {
      int mid = low + (high - low) / 2;

      if (arr[mid] == x)
        return mid;
        if (arr[mid] < x)
            low = mid + 1;

        else
            high = mid - 1;
    }

    return -1;
  }

  public static void main(String args[])
  {
    BinarySearch ob = new BinarySearch();
    int arr[] = { 2, 3, 4, 10, 40 };
    int n = arr.length;
    int x = 10;
    int result = ob.binarySearch(arr, x);
    if (result == -1)
        System.out.println(
                "Element is not present in array");
    else
        System.out.println("Element is present at "
                + "index " + result);
  }
}
```

OUTPUT:



```java
class BinarySearch {

    int binarySearch(int arr[], int x)
    {
        int low = 0, high = arr.length - 1;
        while (low <= high) {
            int mid = low + (high - low) / 2;

            if (arr[mid] == x)
                return mid;

            if (arr[mid] < x)
                low = mid + 1;

            else
                high = mid - 1;
        }
```

```
C:\Users\sanja\.jdks\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\sanja\AppData\Local\JetBrains\IntelliJ IDEA 2024.2.4\lib\idea_rt.jar=56738:C:\Users\sanja\AppData\Local\Je
Element is present at index 3

Process finished with exit code 0
```

6.Next Greater Element:

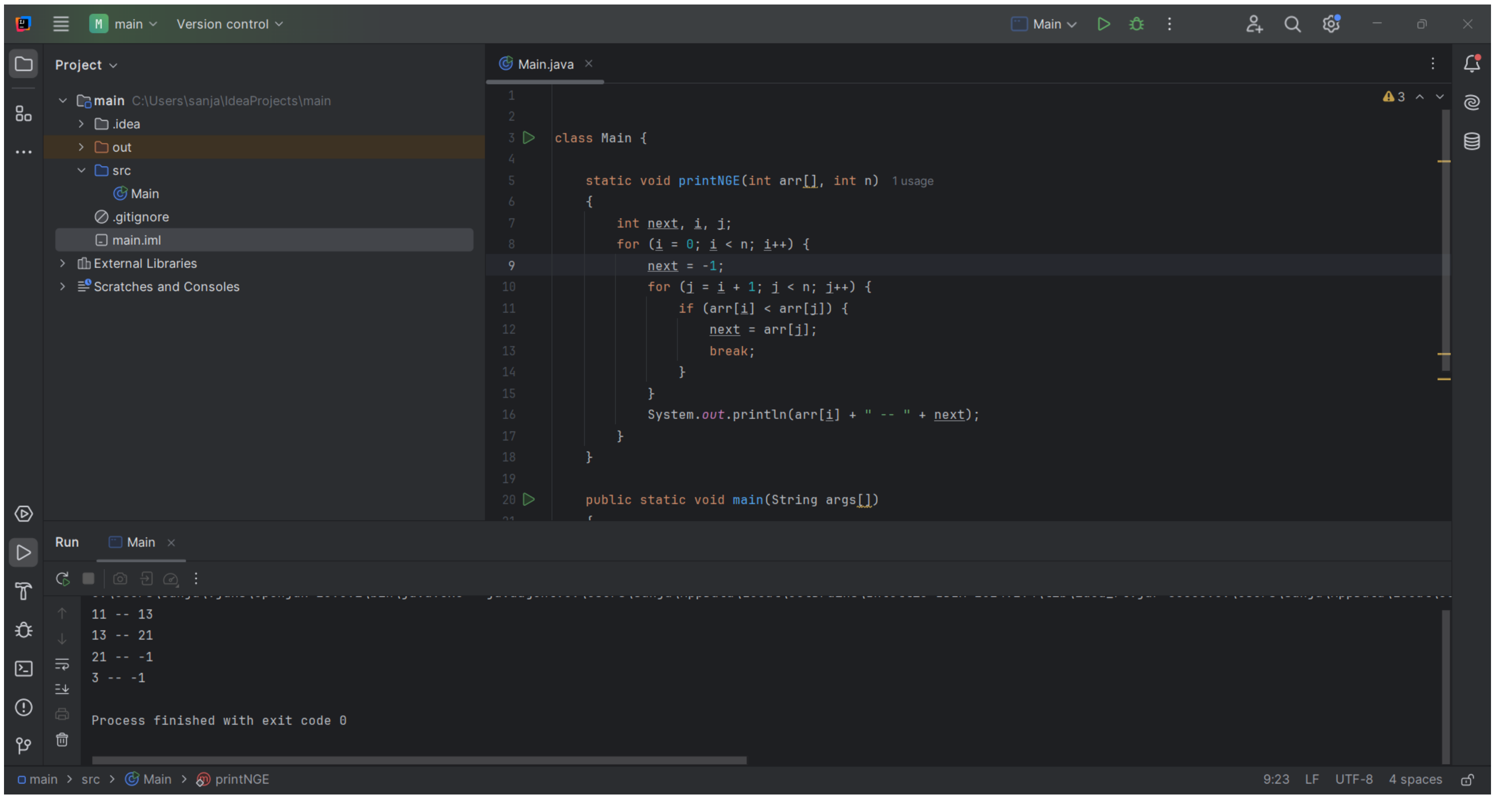
Time complexity:O(n2)

SOLUTION:

```
class Main {

  static void printNGE(int arr[], int n)
  {
    int next, i, j;
    for (i = 0; i < n; i++) {
      next = -1;
```

```
            for (j = i + 1; j < n; j++) {
                if (arr[i] < arr[j]) {
                    next = arr[j];
                    break;
                }
            System.out.println(arr[i] + " -- " + next);
        }
    }

    public static void main(String args[])
    {
        int arr[] = { 11, 13, 21, 3 };
        int n = arr.length;
        printNGE(arr, n);
    }
}
```

OUTPUT:



```java
class Main {

    static void printNGE(int arr[], int n)
    {
        int next, i, j;
        for (i = 0; i < n; i++) {
            next = -1;
            for (j = i + 1; j < n; j++) {
                if (arr[i] < arr[j]) {
                    next = arr[j];
                    break;
                }
            }
            System.out.println(arr[i] + " -- " + next);
        }
    }

    public static void main(String args[])
```

```
11 -- 13
13 -- 21
21 -- -1
3 -- -1

Process finished with exit code 0
```

7. Union Of Two Arrays:

Time Complexity: O(n*m)

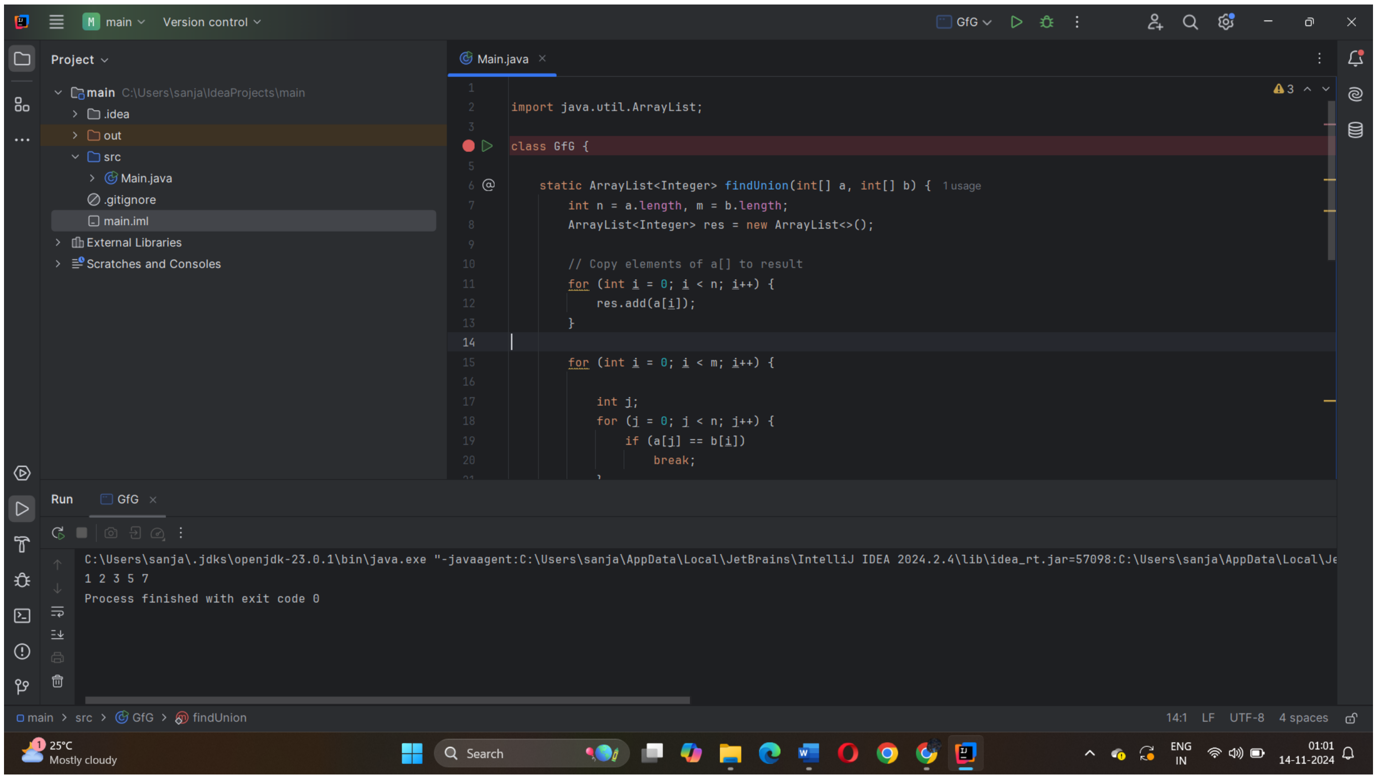
SOLUTION:

```java
import java.util.ArrayList;

class GfG {

  static ArrayList<Integer> findUnion(int[] a, int[] b) {
     int n = a.length, m = b.length;
     ArrayList<Integer> res = new ArrayList<>();
```

```
    for (int i = 0; i < n; i++) {
        res.add(a[i]);
    }

    for (int i = 0; i < m; i++) {

        int j;
        for (j = 0; j < n; j++) {
            if (a[j] == b[i])
                break;
        }

        if (j == n) {
            res.add(b[i]);
        }
    }

    res.sort(null);
    return res;
}

public static void main(String[] args) {
    int[] a = {1, 2, 3};
    int[] b = {2, 5, 7};
```

```
        ArrayList<Integer> res = findUnion(a, b);

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

OUTPUT:



```

import java.util.ArrayList;

class GfG {

    static ArrayList<Integer> findUnion(int[] a, int[] b) {
        int n = a.length, m = b.length;
        ArrayList<Integer> res = new ArrayList<>();

        // Copy elements of a[] to result
        for (int i = 0; i < n; i++) {
            res.add(a[i]);
        }

        for (int i = 0; i < m; i++) {

            int j;
            for (j = 0; j < n; j++) {
                if (a[j] == b[i])
                    break;
```

```
C:\Users\sanja\.jdks\openjdk-23.0.1\bin\java.exe "-javaagent:C:\Users\sanja\AppData\Local\JetBrains\IntelliJ IDEA 2024.2.4\lib\idea_rt.jar=57098:C:\Users\sanja\AppData\Local\Je
1 2 3 5 7
Process finished with exit code 0
```