

DSA Practise Set – 1

Date: 10-11-2024

Name: Ramachandran M

Reg No: 22CS124

Dept: Computer Science Engineering

Problem 1:

```
import java.util.*;

public class One {

    public static void main(String[] args){

        int[] nums1 = new int[]{2, 3, -8, 7, -1, 2, 3};

        int ans = func(nums1);

        System.out.println(ans);

        int[] nums2 = new int[]{-2, -4};

        int ans2 = func(nums2);

        System.out.println(ans2);

    }

    public static int func (int[] nums){

        int n = nums.length;

        int sum=0;

        int max = Integer.MIN_VALUE;

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

            sum+=nums[i];

            if(sum>max){

                max=sum;

            }

            if(sum<0){

                sum=0;

            }

        }

    }

}
```

```

    return max;
}
}

```

The screenshot shows an IDE with a Java file named 'One.java'. The code defines a class 'One' with a 'main' method and a 'func' method. The 'main' method creates two integer arrays, 'nums1' and 'nums2', and calls 'func' on each. The 'func' method iterates through the array to find the maximum value. The terminal shows the output of running 'javac One.java' and 'java One.java', which prints the results of the function calls.

```

1  import java.util.*;
2  public class One {
3      public static void main(String[] args){
4          int[] nums1 = new int[]{2, 3, -8, 7, -1, 2, 3};
5          int ans = func(nums1);
6          System.out.println(ans);
7          int[] nums2 = new int[]{-2, -4};
8          int ans2 = func(nums2);
9          System.out.println(ans2);
10     }
11     public static int func (int[] nums){
12         int n = nums.length;
13         int sum=0;
14         int max = Integer.MIN_VALUE;
15         for(int i=0;i<n;i++){
16             sum+=nums[i];
17             if(sum>max){
18                 max=sum;
19             }
20             if(sum<0){
21                 sum=0;
22             }
23         }
24         return max;
25     }
26 }

```

```

$ javac One.java
$ java One.java
11
-2

```

Problem 2:

```

public class Two {

    public static void main(String[] args){

        int[] nums1 = new int[]{-2, 6, -3, -10, 0, 2};

        int ans1 = func(nums1);

        System.out.println(ans1);

        int[] nums2 = new int[]{-1, -3, -10, 0, 60};

        int ans2 = func(nums2);

        System.out.println(ans2);

    }

    public static int func(int[] nums){

        int n = nums.length;

        int curmaxproduct = nums[0];

        int curminproduct = nums[0];

        int max = nums[0];

        for(int i=1;i<n;i++){

            int temp = curmaxproduct;

```

```

        curmaxproduct =
Math.max(nums[i],Math.max(nums[i]*curmaxproduct,nums[i]*curminproduct));

        curminproduct = Math.min(nums[i],Math.min(nums[i]*temp,nums[i]*curminproduct));

        max = Math.max(max,curmaxproduct);

    }

    return max;

}
}

```

The screenshot shows an IDE with a Java file named 'Two.java'. The code defines a class 'Two' with a 'main' method and a 'func' method. The 'main' method creates two integer arrays, 'nums1' and 'nums2', and calls 'func' on each. The 'func' method calculates the maximum and minimum products of a subarray. The output window shows the results of running the program: '180' and '60'.

```

1 public class Two {
2     public static void main(String[] args){
3         int[] nums1 = new int[]{-2, 6, -3, -10, 0, 2};
4         int ans1 = func(nums1);
5         System.out.println(ans1);
6         int[] nums2 = new int[]{-1, -3, -10, 0, 60};
7         int ans2 = func(nums2);
8         System.out.println(ans2);
9     }
10    public static int func(int[] nums){
11        int n = nums.length;
12        int curmaxproduct = nums[0];
13        int curminproduct = nums[0];
14        int max = nums[0];
15        for(int i=1;i<n;i++){
16            int temp = curmaxproduct;
17            curmaxproduct = Math.max(nums[i],Math.max(nums[i]*curmaxproduct,nums[i]*curminproduct));
18            curminproduct = Math.min(nums[i],Math.min(nums[i]*temp,nums[i]*curminproduct));
19            max = Math.max(max,curmaxproduct);
20        }
21        return max;
22    }
23 }
24

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

de||@ramachandranvuthusamy MINGW64 ~/desktop
$ javac Two.java

de||@ramachandranvuthusamy MINGW64 ~/desktop
$ java Two.java
180
60

```

Ln 19, Col 47 Spaces: 4 UTF-8 CRLF Java Go Live

Problem 3:

```

public class Three {

    public static void main(String[] args){

        int[] nums1 = new int[] {4, 5, 6, 7, 0, 1, 2};

        int key1 = 0;

        int val1 = func(nums1,key1);

        int[] nums2 = new int[] {4, 5, 6, 7, 0, 1, 2};

        int key2 = 3;

        int val2 = func(nums2,key2);

        int[] nums3 = new int[] {50, 10, 20, 30, 40};

        int key3 = 10;

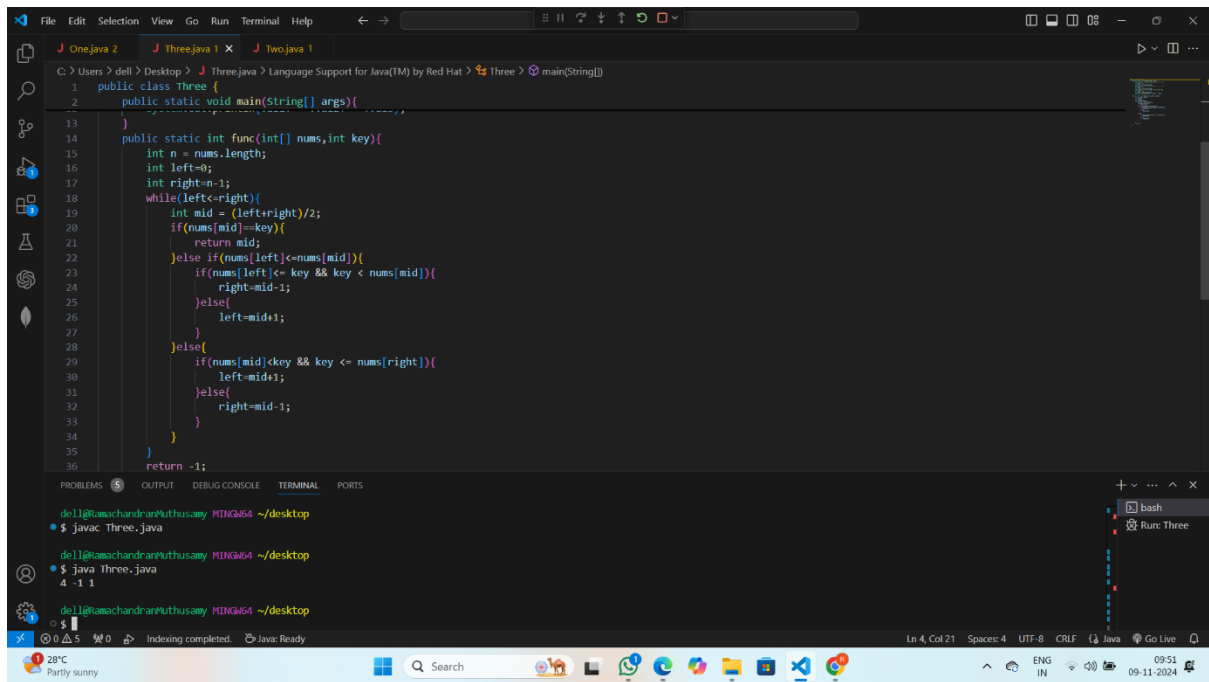
        int val3 = func(nums3,key3);
    }
}

```

```

        System.out.println(val1+" "+val2+" "+val3);
    }
    public static int func(int[] nums,int key){
        int n = nums.length;
        int left=0;
        int right=n-1;
        while(left<=right){
            int mid = (left+right)/2;
            if(nums[mid]==key){
                return mid;
            }else if(nums[left]<=nums[mid]){
                if(nums[left]<= key && key < nums[mid]){
                    right=mid-1;
                }else{
                    left=mid+1;
                }
            }else{
                if(nums[mid]<key && key <= nums[right]){
                    left=mid+1;
                }else{
                    right=mid-1;
                }
            }
        }
        return -1;
    }
}

```



Problem 4:

```

public class Four {

    public static void main(String[] args){

        int[] nums1 = new int[]{1, 5, 4, 3};

        int ans1 = func(nums1);

        int[] nums2 = new int[]{3, 1, 2, 4, 5};

        int ans2 = func(nums2);

        System.out.println(ans1+" "+ans2);

    }

    public static int func(int[] nums){

        int n = nums.length;

        int max=0;

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

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

                int min = Math.min(nums[i],nums[j]);

                int len = Math.abs(i-j);

                int area = min*len;

                max = Math.max(max,area);

            }

        }

    }

}

```

```

    return max;
}
}

```

The screenshot shows an IDE with a Java file named 'Four.java'. The code defines a class 'Four' with a 'main' method and a 'func' method. The 'main' method creates two integer arrays, 'nums1' and 'nums2', and calls 'func' on each. The 'func' method calculates the maximum value of the product of the minimum and maximum elements of an array. The terminal shows the command 'javac Four.java' and 'java Four.java' being executed, resulting in the output '6 12'.

```

1 public class Four {
2     public static void main(String[] args){
3         int[] nums1 = new int[]{1, 5, 4, 3};
4         int ans1 = func(nums1);
5         int[] nums2 = new int[]{3, 1, 2, 4, 5};
6         int ans2 = func(nums2);
7         System.out.println(ans1+" "+ans2);
8     }
9     public static int func(int[] nums){
10        int n = nums.length;
11        int max=0;
12        for(int i=0;i<n;i++){
13            for(int j=i+1;j<n;j++){
14                int min = Math.min(nums[i],nums[j]);
15                int len = Math.abs(i-j);
16                int area = min*len;
17                max = Math.max(max,area);
18            }
19        }
20        return max;
21    }
22 }
23

```

```

del@kamachandranvuthusamy MINKA64 ~/desktop
$ javac Four.java

del@kamachandranvuthusamy MINKA64 ~/desktop
$ java Four.java
6 12

```

Problem 5:

```

import java.math.BigInteger;

public class Five {

    public static void main(String[] args){

        int n = 100;

        BigInteger ans = func(BigInteger.valueOf(n));

        int m = 50;

        BigInteger ans2 = func(BigInteger.valueOf(m));

        System.out.println(ans+" "+ans2);

    }

    public static BigInteger func(BigInteger n){

        if(n.equals(BigInteger.ONE)){

            return BigInteger.ONE;

        }

        return n.multiply(func(n.subtract(BigInteger.ONE)));

    }

}

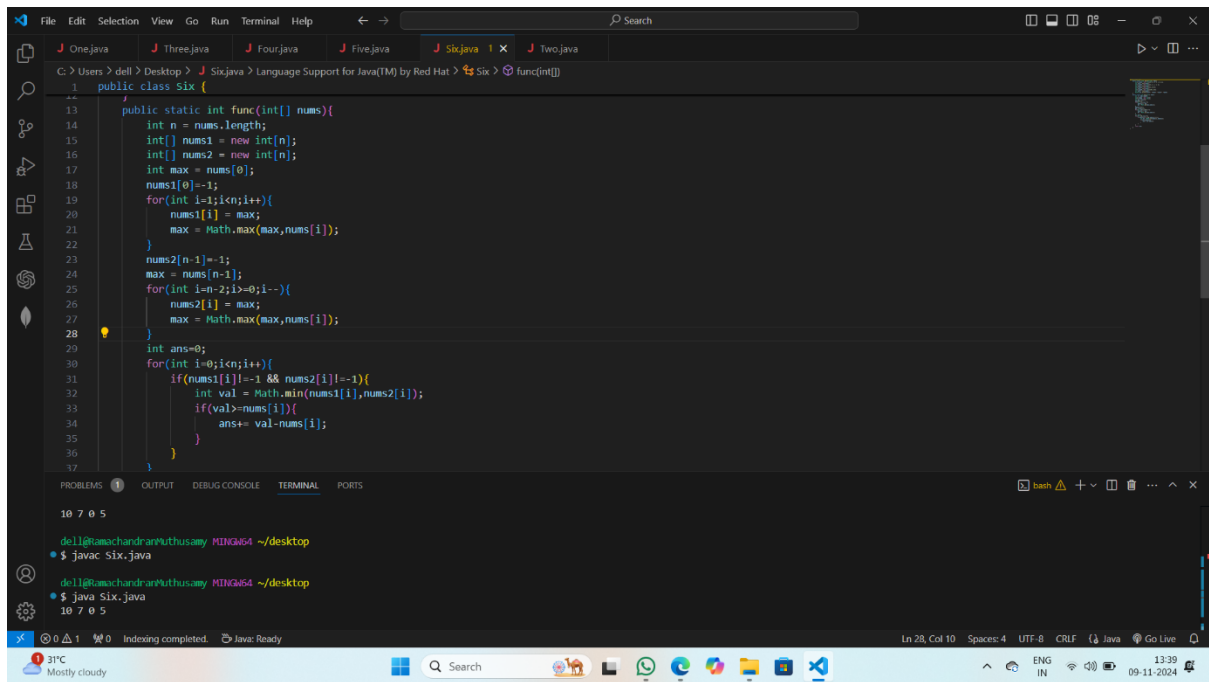
```



```

        nums1[i] = max;
        max = Math.max(max,nums[i]);
    }
    nums2[n-1]=-1;
    max = nums[n-1];
    for(int i=n-2;i>=0;i--){
        nums2[i] = max;
        max = Math.max(max,nums[i]);
    }
    int ans=0;
    for(int i=0;i<n;i++){
        if(nums1[i]!=-1 && nums2[i]!=-1){
            int val = Math.min(nums1[i],nums2[i]);
            if(val>=nums[i]){
                ans+= val-nums[i];
            }
        }
    }
    return ans;
}
}

```

Problem 7:

import java.util.*;

public class Seven {

public static void main(String[] args){

int[] nums1 = new int[]{7, 3, 2, 4, 9, 12, 56};

int m1=3;

int[] nums2 = new int[]{7, 3, 2, 4, 9, 12, 56};

int m2 = 5;

int ans1 = func(nums1,m1);

int ans2 = func(nums2,m2);

System.out.println(ans1+" "+ans2);

}

public static int func(int[] nums , int stud){

int n = nums.length;

Arrays.sort(nums);

int min = Integer.MAX_VALUE;

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

min = Math.min(min,nums[i+stud-1]-nums[i]);

}

return min;

```

}

}

```

```

1  import java.util.*;
2  public class Seven {
3      public static void main(String[] args){
4          int[] nums1 = new int[]{7, 3, 2, 4, 9, 12, 56};
5          int m1=3;
6          int[] nums2 = new int[]{7, 3, 2, 4, 9, 12, 56};
7          int m2 = 5;
8          int ans1 = func(nums1,m1);
9          int ans2 = func(nums2,m2);
10         System.out.println(ans1+" "+ans2);
11     }
12     public static int func(int[] nums , int stud){
13         int n = nums.length;
14         Arrays.sort(nums);
15         int min = Integer.MAX_VALUE;
16         for(int i=0;i<n-stud;i++){
17             min = Math.min(min,nums[i+stud]-nums[i]);
18         }
19         return min;
20     }
21 }
22

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS

```

de||@RamachandranVithusamy MINGW64 ~/desktop
$ javac Seven.java

de||@RamachandranVithusamy MINGW64 ~/desktop
$ java Seven.java
2 7

```

Ln 9, Col 35 Spaces: 4 UTF-8 CRLF Java Go Live

Problem 8:

```

import java.util.*;

public class Eight {

    public static void main(String[] args){

        int[][] nums1 = new int[][] {{1, 3}, {2, 4}, {6, 8}, {9, 10}};

        int[][] ans1 = func(nums1);

        int[][] nums2 = new int[][]{{7, 8}, {1, 5}, {2, 4}, {4, 6}};

        int[][] ans2 = func(nums2);

        for(int[] temp : ans1){

            for(int no : temp){

                System.out.print(no+" ");

            }

            System.out.print(",");

        }

        System.out.println();

        for(int[] temp : ans2){

```

```

        for(int no : temp){
            System.out.print(no+" ");
        }
        System.out.print(",");
    }
}

public static int[][] func(int[][] nums){
    int n = nums.length;
    List<Integer> ls = new ArrayList<>();
    Arrays.sort(nums,(a,b)->Integer.compare(a[0],b[0]));
    int prevstart = nums[0][0];
    int prevend = nums[0][1];
    for(int i=1;i<n;i++){
        int curstart = nums[i][0];
        int curend = nums[i][1];
        if(curstart<=prevend){
            prevend = curend;
        }else{
            ls.add(prevstart);
            ls.add(prevend);
            prevstart = curstart;
            prevend = curend;
        }
    }
    ls.add(prevstart);
    ls.add(prevend);
    int len = (ls.size())/2;
    System.out.println(len);
    int[][] ans = new int[len][2];
    for(int i=0;i<len;i++){
        ans[i][0] = ls.get(0);
        ls.remove(0);
    }
}

```

```

        ans[i][1] = ls.get(0);

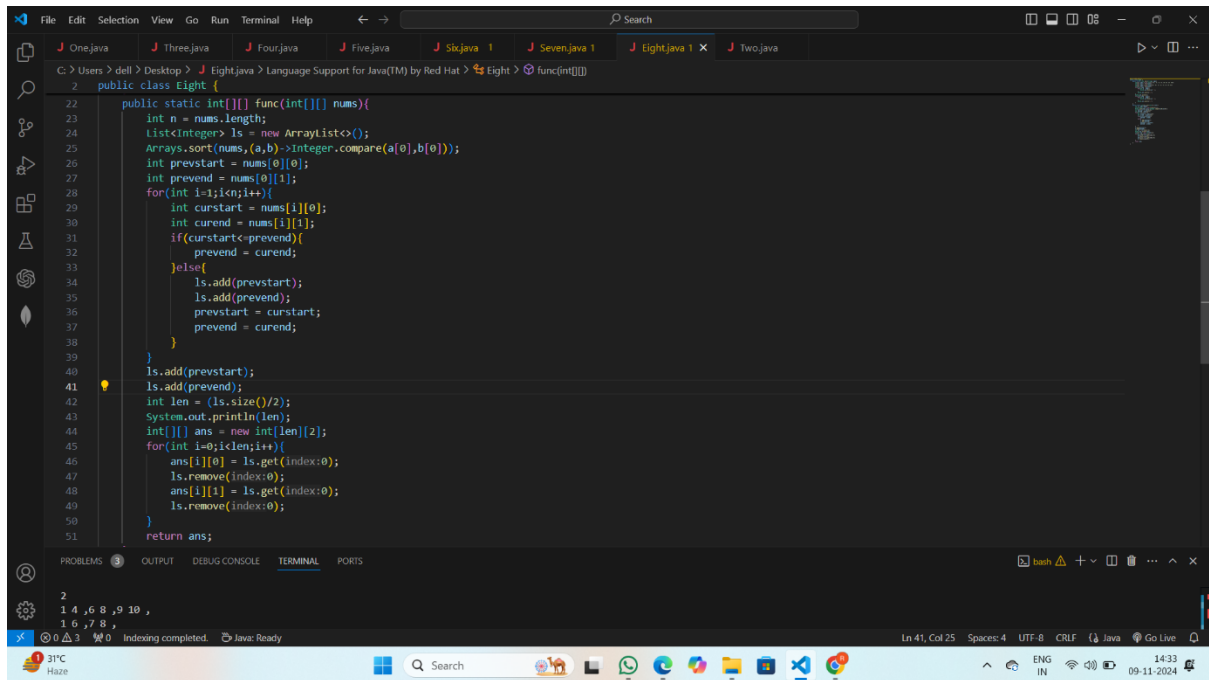
        ls.remove(0);

    }

    return ans;

}
}

```



Problem 9:

```

import java.util.*;

public class Nine {

    public static void main(String[] args){

        int[][] mat1 = new int[][]{{1, 0},{0, 0}};

        func(mat1);

        int[][] mat2 = new int[][]{{0, 0, 0},{0, 0, 1}};

        func(mat2);

        int[][] mat3 = new int[][]{{1, 0, 0, 1},{0, 0, 1, 0},{0, 0, 0, 0}};

        func(mat3);

    }

    public static void func(int[][] nums){

        int n = nums.length;

        int m = nums[0].length;
    }
}

```

```

List<Integer> ls1 = new ArrayList<>();
List<Integer> ls2 = new ArrayList<>();
for(int i=0;i<n;i++){
    for(int j=0;j<m;j++){
        if(nums[i][j]==1){
            ls1.add(i);
            ls2.add(j);
        }
    }
}
for(int i=0;i<n;i++){
    for(int j=0;j<m;j++){
        if(ls1.contains(i) || ls2.contains(j)){
            nums[i][j] = 1;
        }
    }
}
for(int[] temp : nums){
    for(int no : temp){
        System.out.print(no+" ");
    }
    System.out.println();
}
System.out.println();
}
}

```

```

1 public class Nine {
2
3     public static void func(int[][] nums){
4         int n = nums.length;
5         int m = nums[0].length;
6         List<Integer> ls1 = new ArrayList<>();
7         List<Integer> ls2 = new ArrayList<>();
8         for(int i=0;i<n;i++){
9             for(int j=0;j<m;j++){
10                 if(nums[i][j]==1){
11                     ls1.add(i);
12                     ls2.add(j);
13                 }
14             }
15         }
16         for(int i=0;i<n;i++){
17             for(int j=0;j<m;j++){
18                 if(ls1.contains(i) || ls2.contains(j)){
19                     nums[i][j] = 1;
20                 }
21             }
22         }
23         for(int[] temp : nums){
24             for(int no : temp){
25                 System.out.print(no+" ");
26             }
27         }
28     }
29 }

```

```

$ java nine.java
1 1
1 0

0 0 1
1 1 1

1 1 1 1
1 1 1 1
1 0 1 1

```

Problem 10:

import java.util.*;

public class Ten {

public static void main(String[] args){

int[][] mat1 = new int[][] {{1, 2, 3, 4},{5, 6, 7, 8},{9, 10, 11, 12},{13, 14, 15, 16 }};

func(mat1);

int[][] mat2 = new int[][]{{1, 2, 3, 4, 5, 6},{7, 8, 9, 10, 11, 12},{13, 14, 15, 16, 17, 18}};

func(mat2);

}

public static void func(int[][] nums){

int top = 0;

int bottom = nums.length-1;

int left=0;

int right = nums[0].length-1;

List<Integer> ls = new ArrayList<>();

while(left<=right && top<=bottom){

for(int i=left;i<=right;i++){

ls.add(nums[top][i]);

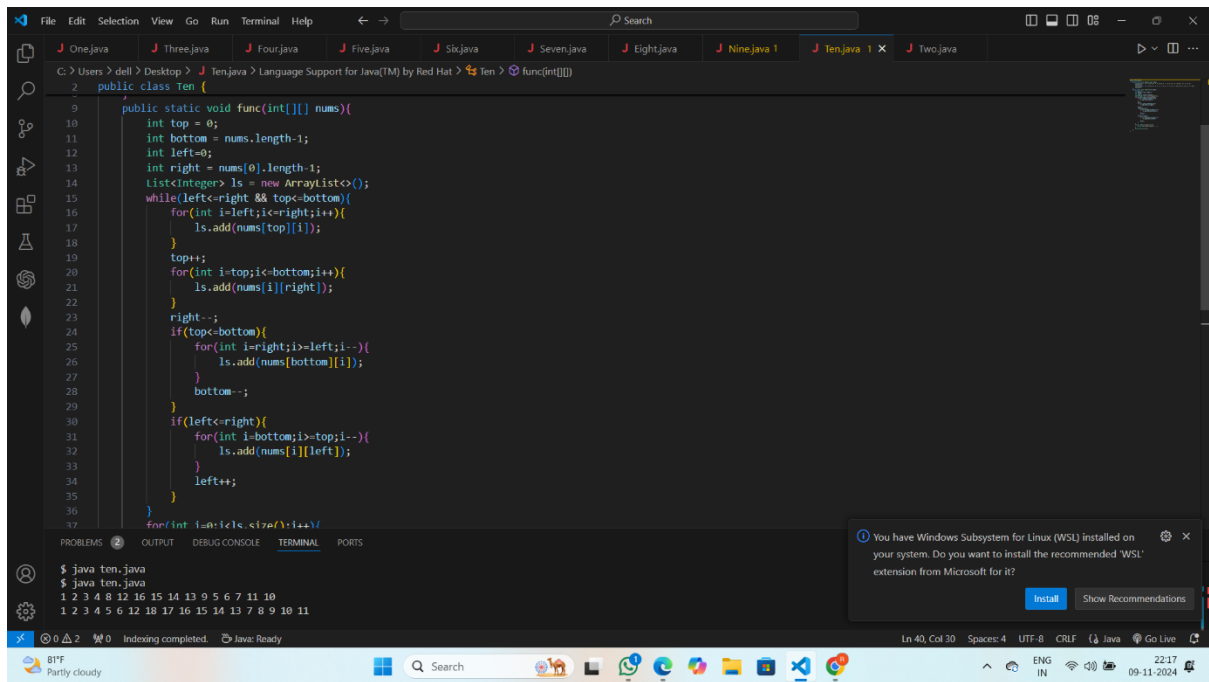
}

top++;

```

        for(int i=top;i<=bottom;i++){
            ls.add(nums[i][right]);
        }
        right--;
        if(top<=bottom){
            for(int i=right;i>=left;i--){
                ls.add(nums[bottom][i]);
            }
            bottom--;
        }
        if(left<=right){
            for(int i=bottom;i>=top;i--){
                ls.add(nums[i][left]);
            }
            left++;
        }
    }
    for(int i=0;i<ls.size();i++){
        System.out.print(ls.get(i)+ " ");
    }
    System.out.println();
}
}

```



Problem 13:

```

import java.util.*;

public class Thirteen {

    public static void main(String[] args){

        String temp1 = "((( ))) ( )";

        func(temp1);

        String temp2 = " ( ) ((( )))";

        func(temp2);

    }

    public static void func(String temp){

        Stack<Character> stk = new Stack<>();

        boolean status = true;

        for(char ch : temp.toCharArray()){

            if(ch=='('){

                stk.push(ch);

            }else{

                if(!stk.isEmpty()){

                    stk.pop();

                }else{

                    status=false;

                }

            }

        }

    }
  
```



```

        break;
    }
}
}
if(status){
    System.out.println("Balanced");
}else{
    System.out.println("Not Balanced");
}
}
}

```

```

C:\Users> del > Desktop > J. Eleven.java > Language Support for Java(TM) by Red Hat > Eleven > func(String)
2 public class Eleven {
3     public static void main(String[] args){
4
5     }
6
7     public static void func(String temp){
8         stack<Character> stk = new Stack<>();
9         boolean status = true;
10        for(char ch : temp.toCharArray()){
11            if(ch=='('){
12                stk.push(ch);
13            }else{
14                if(!stk.isEmpty()){
15                    stk.pop();
16                }else{
17                    status=false;
18                    break;
19                }
20            }
21        }
22    }
23
24    if(status){
25        System.out.println(x:"Balanced");
26    }else{
27        System.out.println(x:"Not Balanced");
28    }
29 }
30 }
31

```

```

del@hamachandranuthusamy MINGW64 ~/desktop
$ java eleven.java
Balanced
Not Balanced

```

Problem 14:

```

import java.util.*;

public class Fourteen {

    public static void main(String[] args){

        String s1 = "geeks";

        String s2 = "kseeg";

        func(s1,s2);

        String s3 = "allergic";
    }
}

```

```

String s4 = "allergy";
func(s3,s4);
String s5 = "g";
String s6 = "g";
func(s5,s6);
}
public static void func(String s1 , String s2){
    Map<Character,Integer> map1 = new HashMap<>();
    Map<Character,Integer> map2 = new HashMap<>();
    boolean status = true;
    if(s1.length()!=s2.length()){
        System.out.println(false);
        return;
    }
    for(int i=0;i<s1.length();i++){
        map1.put(s1.charAt(i),map1.getOrDefault(s1.charAt(i),0)+1);
        map2.put(s2.charAt(i),map2.getOrDefault(s2.charAt(i),0)+1);
    }
    for(Map.Entry<Character,Integer> entry : map1.entrySet()){
        Character key1 = entry.getKey();
        int val1 = entry.getValue();
        int val2 = map2.get(key1);
        if(val1!=val2){
            status=false;
            break;
        }
    }
    System.out.println(status);
}
}

```

```

1  public class Fourteen {
2      public static void main(String[] args){
3
4          String s3 = "allergic";
5          String s4 = "allergy";
6          func(s3,s4);
7          String s5 = "g";
8          String s6 = "g";
9          func(s5,s6);
10     }
11     public static void func(String s1 , String s2){
12         Map<Character,Integer> map1 = new HashMap<>();
13         Map<Character,Integer> map2 = new HashMap<>();
14         boolean status = true;
15         if(s1.length()!=s2.length()){
16             System.out.println(x:false);
17             return;
18         }
19         for(int i=0;i<s1.length();i++){
20             map1.put(s1.charAt(i),map1.getOrDefault(s1.charAt(i),defaultValue:0)+1);
21             map2.put(s2.charAt(i),map2.getOrDefault(s2.charAt(i),defaultValue:0)+1);
22         }
23         for(Map.Entry<Character,Integer> entry : map1.entrySet()){
24             Character key1 = entry.getKey();
25             int val1 = entry.getValue();
26             int val2 = map2.get(key1);
27             if(val1!=val2){
28                 status=false;
29             }
30         }
31     }
32 }

```

```

dell@ramachandranvuthusamy MINGW64 ~/desktop
$ java fourteen.java
true
false
true

```

Problem 15:

```

public class fifteen {

    public static void main(String[] args){

        String s1 = "forgeeksskeegfor";

        func(s1);

        String s2 = "geeks";

        func(s2);

        String s3 = "abc";

        func(s3);

    }

    public static void func(String s1){

        int n = s1.length();

        int start = 0;

        int maxlen = 1;

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

            int len1 = find(s1,i,i);

            int len2 = find(s1,i,i+1);

            int len = Math.max(len1,len2);

            if(len>maxlen){

                maxlen=len;

```

```

        start=i-(len-1)/2;
    }
}

System.out.println(s1.substring(start,start+maxlen));
}

public static int find(String s1 , int left,int right){
    while(left>=0 && right<=s1.length()-1 && s1.charAt(left)==s1.charAt(right)){
        left--;
        right++;
    }
    return right-left-1;
}
}

```

The screenshot shows an IDE with a Java file named 'fifteen.java'. The code defines a 'find' method that takes a string 's1' and two integers 'left' and 'right'. It uses a while loop to expand from the center outwards until characters no longer match. The 'main' method calls 'find' for each character in the string 'a' and prints the maximum length of a palindromic substring. The terminal shows the command 'java fifteen.java' being executed, resulting in the output 'a'.

```

1  public class fifteen {
2      public static void main(String[] args){
3          // ...
4          // ...
5          // ...
6          // ...
7          // ...
8          // ...
9          // ...
10         // ...
11         // ...
12         // ...
13         // ...
14         // ...
15         // ...
16         // ...
17         // ...
18         // ...
19         // ...
20         // ...
21         // ...
22         // ...
23         // ...
24         // ...
25         // ...
26         // ...
27         // ...
28         // ...
29         // ...
30         // ...
31         // ...
32         // ...
33         // ...
34     }
35 }

```

```

1  public static void func(String s1){
2      int n = s1.length();
3      int start = 0;
4      int maxlen = 1;
5      for(int i=0;i<n;i++){
6          int len1 = find(s1,i,i);
7          int len2 = find(s1,i,i+1);
8          int len = Math.max(len1,len2);
9          if(len>maxlen){
10             maxlen=len;
11             start=i-(len-1)/2;
12         }
13     }
14     System.out.println(s1.substring(start,start+maxlen));
15 }
16
17 public static int find(String s1 , int left,int right){
18     while(left>=0 && right<=s1.length()-1 && s1.charAt(left)==s1.charAt(right)){
19         left--;
20         right++;
21     }
22     return right-left-1;
23 }
24 }

```

```

dell@ramachandranvuthusamy MINGW64 ~/desktop
$ java fifteen.java
a

```

Problem 16:

```

import java.util.*;

public class Sixteen {

    public static void main(String[] args){

        String[] arr = new String[]{"geeksforgeeks", "geeks", "geek", "geeze"};

        func(arr);
    }
}

```

```

String[] arr2 = new String[]{"hello","world"};

func(arr2);
}

public static void func(String[] arr){

    int n = arr.length;

    Arrays.sort(arr);

    int i=0;

    int firstlen = arr[0].length();

    int lastlen = arr[arr.length-1].length();

    while(i<firstlen && i<lastlen && arr[0].charAt(i)==arr[arr.length-1].charAt(i)){

        i++;

    }

    System.out.println(i!=0?arr[0].substring(0,i):-1);

}
}

```

```

C:\Users> del > Desktop > J Sixteen.java > Language Support for Java(TM) by Red Hat > Sixteen > func(String[])
1  import java.util.*;
2  public class Sixteen {
3      public static void main(String[] args){
4          String[] arr = new String[]{"geeksforgeeks", "geeks", "geek", "geeze"};
5          func(arr);
6          String[] arr2 = new String[]{"hello","world"};
7          func(arr2);
8      }
9      public static void func(String[] arr){
10         int n = arr.length;
11         Arrays.sort(arr);
12         int i=0;
13         int firstlen = arr[0].length();
14         int lastlen = arr[arr.length-1].length();
15         while(i<firstlen && i<lastlen && arr[0].charAt(i)==arr[arr.length-1].charAt(i)){
16             i++;
17         }
18         System.out.println(i!=0?arr[0].substring(beginIndex:0,i):-1);
19     }
20 }
21

```

```

de@l@amachandranruthusamy MINGW64 ~/desktop
$ java sixteen.java
gee
-1

```

Problem 17:

```

import java.util.*;

public class Seventeen {

    public static void main(String[] args){

```

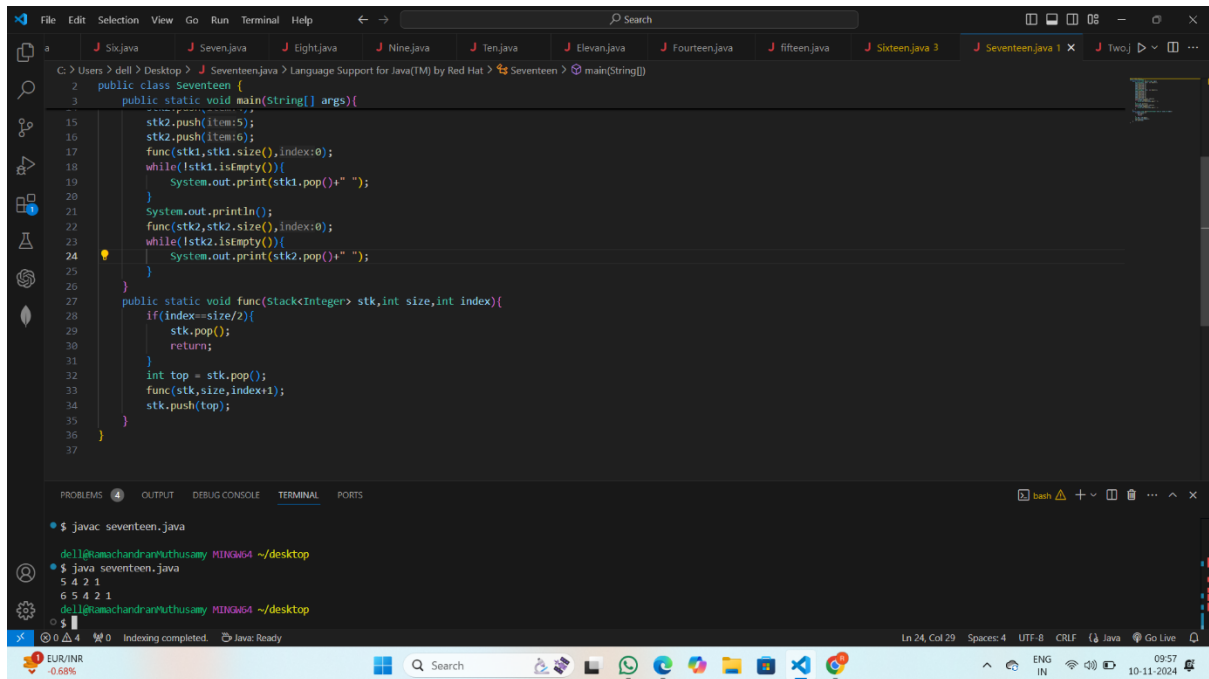
```

Stack<Integer> stk1 = new Stack<>();
stk1.push(1);
stk1.push(2);
stk1.push(3);
stk1.push(4);
stk1.push(5);
Stack<Integer> stk2 = new Stack<>();
stk2.push(1);
stk2.push(2);
stk2.push(3);
stk2.push(4);
stk2.push(5);
stk2.push(6);
func(stk1,stk1.size(),0);
while(!stk1.isEmpty()){
    System.out.print(stk1.pop()+" ");
}
System.out.println();
func(stk2,stk2.size(),0);
while(!stk2.isEmpty()){
    System.out.print(stk2.pop()+" ");
}
}

public static void func(Stack<Integer> stk,int size,int index){
    if(index==size/2){
        stk.pop();
        return;
    }
    int top = stk.pop();
    func(stk,size,index+1);
    stk.push(top);
}

```

}



```
C:\Users> cd Desktop > cd Seventeen > Language Support for Java(TM) by Red Hat > Seventeen > main(String[])
2 public class Seventeen {
3     public static void main(String[] args){
4
5
6
7         stk2.push(item:5);
8         stk2.push(item:6);
9         func(stk1,stk1.size(),index:0);
10        while(!stk1.isEmpty()){
11            System.out.print(stk1.pop()+" ");
12        }
13        System.out.println();
14        func(stk2,stk2.size(),index:0);
15        while(!stk2.isEmpty()){
16            System.out.print(stk2.pop()+" ");
17        }
18    }
19    public static void func(Stack<Integer> stk,int size,int index){
20        if(index==size/2){
21            stk.pop();
22            return;
23        }
24        int top = stk.pop();
25        func(stk,size,index+1);
26        stk.push(top);
27    }
28 }
29
30
31
32
33
34
35
36
37
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
$ javac seventeen.java
de||@kamachandranvithusamy MINKA64 ~/desktop
$ java seventeen.java
5 4 2 1
de||@kamachandranvithusamy MINKA64 ~/desktop
$
```

Problem 18:

```
public class Eighteen {

    public static void main(String[] args){

        int[] nums1 = new int[]{4,5,2,25};

        func(nums1);

        int[] nums2 = new int[]{ 13 , 7, 6 , 12};

        func(nums2);

    }

    public static void func(int[] nums){

        int n = nums.length;

        int[] ans = new int[n];

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

            for(int j=i+1;j<n;j++){

                if(nums[j]>nums[i]){

                    ans[i] = nums[j];

                    break;

                }

            }

        }

    }

}
```

```

        if(ans[i]==0){
            ans[i]=-1;
        }
    }
    for(int no : ans){
        System.out.print(no==0?-1:no+" ");
    }
    System.out.println();
}
}

```

The screenshot shows an IDE with a project named 'Eighteen'. The main file 'Eighteen.java' contains the following code:

```

1 public class Eighteen {
2     public static void main(String[] args){
3         int[] nums1 = new int[]{4,5,2,25};
4         func(nums1);
5         int[] nums2 = new int[]{ 13 , 7, 6 , 12};
6         func(nums2);
7     }
8     public static void func(int[] nums){
9         int n = nums.length;
10        int[] ans = new int[n];
11        for(int i=0;i<n;i++){
12            for(int j=i+1;j<n;j++){
13                if(nums[j]>nums[i]){
14                    ans[i] = nums[j];
15                    break;
16                }
17            }
18            if(ans[i]==0){
19                ans[i]=-1;
20            }
21        }
22        for(int no : ans){
23            System.out.print(no==0?-1:no+" ");
24        }
25        System.out.println();
26    }
27 }

```

The terminal output shows the execution of the program:

```

dell@ramachandranvuthusamy MINK654 ~/desktop
$ java eighteen.java
5 25 25 -1
-1 12 12 -1

```

Problem 19:

```

import java.util.*;

public class Nineteen {

    public static void main(String[] args){

        Node root = new Node(1);

        root.left=new Node(2);

        root.right=new Node(3);

        root.left.left= new Node(4);

        root.left.right=new Node(5);
    }
}

```



```

        root.right.left=new Node(6);
        root.right.right=new Node(7);
        List<Integer> ans = func(root);
        System.out.println(ans);
    }

    public static List<Integer> func(Node root){
        List<Integer> ls = new ArrayList<>();
        int[] maxlevel = new int[]{-1};
        int level = 0;
        find(root,level,maxlevel,ls);
        return ls;
    }

    public static void find(Node root,int level , int[] maxlevel,List<Integer> ls){
        if(root==null){
            return;
        }
        if(level>maxlevel[0]){
            ls.add(root.data);
            maxlevel[0] = level;
        }
        if(root.right!=null){
            find(root.right,level+1,maxlevel,ls);
        }
        if(root.left!=null){
            find(root.left,level+1,maxlevel,ls);
        }
    }

}

class Node{
    int data;
    Node left;

```

Node right;

Node(int x){

 this.data = x;

 left = null;

 right = null;

}

}

The screenshot shows an IDE with a Java file named Nineteen.java. The code defines a Node class and a recursive function to find the maximum level of a binary tree. The terminal window shows the execution of the program, which outputs the maximum level as 3.

```
File Edit Selection View Go Run Terminal Help
C:\Users> del > Desktop > J Nineteen.java > Language Support for Java(TM) by Red Hat > Nineteen > find(Node, int, int[], List<Integer>)
2 public class Nineteen {
3     public static void main(String[] args){
13
14     public static List<Integer> func(Node root){
15         List<Integer> ls = new ArrayList<>();
16         int[] maxlevel = new int[1];
17         int level = 0;
18         find(root, level, maxlevel, ls);
19         return ls;
20     }
21     public static void find(Node root, int level, int[] maxlevel, List<Integer> ls){
22         if(root==null){
23             return;
24         }
25         if(level>maxlevel[0]){
26             ls.add(root.data);
27             maxlevel[0] = level;
28         }
29         if(root.right!=null){
30             find(root.right, level+1, maxlevel, ls);
31         }
32         if(root.left!=null){
33             find(root.left, level+1, maxlevel, ls);
34         }
35     }
36 }
37 }

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
$ javac nineteen.java
del@RamachandranVithusamy MINGW64 ~/desktop
$ java nineteen.java
[1, 3, 7]
del@RamachandranVithusamy MINGW64 ~/desktop
$
```

Problem 20:

public class Twenty {

 public static void main(String[] args){

 Node root = new Node(1);

 root.left=new Node(2);

 root.right=new Node(3);

 root.right.left=new Node(6);

 root.right.right=new Node(7);

 System.out.println(func(root));

 }

 public static int func(Node root){

 if(root==null){

 return 0;

```

    }

    int llen = func(root.left);

    int rlen = func(root.right);

    return Math.max(llen,rlen)+1;

}

}

class Node{

    int data;

    Node left;

    Node right;

    Node(int x){

        this.data = x;

        right=left=null;

    }

}

```

The screenshot shows an IDE with a Java file named 'Twenty.java'. The code defines a 'Node' class with 'data', 'left', and 'right' attributes, and a 'Twenty' class with a 'main' method and a 'func' static method. The 'main' method creates a binary tree structure and prints the result of 'func(root)'. The 'func' method calculates the height of the tree. Below the code editor, a terminal window shows the execution of 'javac twenty.java' and 'java twenty.java', resulting in the output '3'.

```

C:\Users> dell > Desktop > J Twenty.java > Language Support for Java(TM) by Red Hat > Twenty > func(Node)
1 public class Twenty {
2     Run main | Debug main | Run | Debug
3     public static void main(String[] args){
4         Node root = new Node(x:1);
5         root.left=new Node(x:2);
6         root.right=new Node(x:3);
7         root.right.left=new Node(x:6);
8         root.right.right=new Node(x:7);
9         System.out.println(func(root));
10    }
11    public static int func(Node root){
12        if(root==null){
13            return 0;
14        }
15        int llen = func(root.left);
16        int rlen = func(root.right);
17        return Math.max(llen,rlen)+1;
18    }
19    class Node{
20        int data;
21        Node left;
22        Node right;
23        Node(int x){
24            this.data = x;
25            right=left=null;
26        }
27    }
28 }

```

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
$ javac twenty.java
dell@kamachandranvithusamy MINGW64 ~/desktop
$ java twenty.java
3
dell@kamachandranvithusamy MINGW64 ~/desktop
$

```