EduBridge

Rohan S

Java Full Stack_Coding Assessment_20's report

Submitted on Mar 26 2023 23:46:27 IST

263.2 (82%)
scored out of 320

Completed
in the assignment

23
problems attempted out of 23

2.4 / 5
avg. code quality score

Severe Violation
flagged by DoSelect proctoring engine

Test time analysis

55m 53s
time taken for completion

Mar 26 2023 22:49:26 IST
test invite time

Mar 26 2023 22:50:33 IST
test start time

Mar 26 2023 23:46:27 IST
test end time

Performance summary

21
solutions accepted

2
solutions partially accepted

Proctor analysis

0
browser used

0
navigation violation

2
webcam violations

0 min
no test window violation

Solutions

Problem Name	Problem Type	Status	Score
AXYAA Digital-Coding			**100.0** / **100** **(100.00%)**
Guide Jack!	Coding	ACCEPTED	**100.0** / 100
Accenture-Coding			**61.1** / **100** **(61.10%)**
Palindromic Name	Coding	PARTIALLY ACCEPTED	**61.1** / 100
Capgemini-Coding			**82.1** / **100** **(82.10%)**
Second Largest	Coding	PARTIALLY ACCEPTED	**82.1** / 100
Java Full Stack-MCQ			**19.0** / **20** **(95.00%)**
Element is	MCQ	ACCEPTED	**1.0** / 1
Is Not Valid	MCQ	ACCEPTED	**1.0** / 1
Talend: Java Mode	MCQ	ACCEPTED	**1.0** / 1
Do We Use	MCQ	ACCEPTED	**1.0** / 1
Catch (exception E)	MCQ	ACCEPTED	**1.0** / 1
Basic Input Keywords	MCQ	ACCEPTED	**1.0** / 1
Running This Code	MCQ	ACCEPTED	**1.0** / 1
Run The Thread	MCQ	ACCEPTED	**1.0** / 1
Entity subclass	MCQ	ACCEPTED	**1.0** / 1
Output MyClass	MCQ	ACCEPTED	**1.0** / 1
New class java	MCQ	ACCEPTED	**1.0** / 1
Topic On Operators	MCQ	ACCEPTED	**1.0** / 1

Problem Name	Problem Type	Status	Score
Find The Output	MCQ	ACCEPTED	**1.0** / 1
The Code Below	MCQ	ACCEPTED	**1.0** / 1
The Given Code	MCQ	ACCEPTED	**1.0** / 1
Integer Array	MCQ	ACCEPTED	**1.0** / 1
Int X=0	MCQ	ACCEPTED	**1.0** / 1
System Code Segment	MCQ	ACCEPTED	**1.0** / 1
Class And Constructors	MCQ	ACCEPTED	**1.0** / 1
Statement(S) Is/Are Correct	MCQ	ACCEPTED	**1.0** / 1

Technology used

Java

Additional Information

Question	Response
Enrollment Number	EBEON1122691435
Batch Code (Eg : 2022-XXXX)	2022-8357

Detailed Report

Section 1 : AXYAA Digital-Coding

Problem 1 : Guide Jack!

CODING SCORE: **100**

Jack is very fond of coding. He is working on string manipulation these days. While he was researching on the internet about string manipulation he came to know that "Strings are immutable, so we use StringBuilder to manipulate a string and change its content".

He wants to write a function that accepts a string and changes all the vowels in the string to the character 'b'. Guide Jack!

Your task here is to implement a **Java** code based on the following specifications. Note that your code should match the specifications in a precise manner. Consider **default visibility** of classes, data fields and methods are public unless mentioned otherwise.

Specifications

```
class definitions:
  class VowelManipulation:
    manipulateVowels(String str):
      return type: StringBuilder
      visibility: public
```

Task:

class VowelManipulation

Implement the below method for this class:

- **StringBuilder manipulateVowels(String str):** accept the string and change all the vowels in the string to the character 'b'.

Sample Input

```
capgemini
```

Sample Output

```
cbpgbmbnb
```

NOTE

- You can make suitable function calls and use **the RUN CODE** button to check your **main()** method output.

Solution

ACCEPTED SCORE: **100.0** / 100

Powered by do select

Code Quality Analysis

Minor quality violations

Quality score: 2.7

Deep Code Analysis Results

Straightforward approach

No cyclomatic constructs detected.

Low modularity

Some reusable components found.

Low extensibility

Some extensible features detected.

```java
1  import java.io.*;
2  import java.util.*;
3  import java.text.*;
4  import java.math.*;
5  import java.util.regex.*;
6
7  class VowelManipulation {
8    //Write Your Code Here..
9    public StringBuilder manipulateVowels(String str)
10   {
11       StringBuilder sb=new StringBuilder();
12       String vowels="aeiouAEIOU";
13       for(char c:str.toCharArray())
14       {
15           if(vowels.indexOf(c)!=-1)
16           {
17               sb.append('b');
18           }
19           else
20           {
21               sb.append(c);
22           }
23       }
24       return sb;
25   }
26 }
27
28 public class Source {
29      public static void main(String args[] ) throws Exception {
30              /* Enter your code here. Read input from STDIN. Print output to STDOUT */
31              Scanner sc =new Scanner(System.in);
32              String input=sc.nextLine();
33              VowelManipulation v= new VowelManipulation();
34              StringBuilder s=v.manipulateVowels(input);
35              System.out.println(s.toString());
36      }
37 }
```

`Java 8`

Evaluation Details

Test_manipulateVowels_1 *(weight:1)*

Status Passed

Execution time 1.53s

CPU 0s

Memory 1MB

Description Testcase passed!

Test_manipulateVowels_3 *(weight:1)*

Status Passed

Execution time 1.67s

CPU 0s

Memory 1MB

Description Testcase passed!

Test_manipulateVowels_2 *(weight:1)*

Status Passed

Execution time 1.65s

CPU 0s

Memory 1MB

Description Testcase passed!

Sample_TC *(sample)*

Status Passed

Execution time 1.49s

CPU 0s

Memory 1MB

Description Testcase passed!

Test_VowelManipulation *(weight:1)*

Status	Passed
Execution time	1.41s
CPU	0s
Memory	432kB
Description	Testcase passed!

Section 2 : Accenture-Coding

Problem 1 : Palindromic Name

CODING | SCORE: **100**

Problem Statement

Parents recently had a kid and they are a huge fan of palindromic strings. Now their relatives have suggested creating a name for their child from both the parent's first names. Now find a way to create a palindromic name from their parents' name. Take a substring of size at least one from both of their names and concatenate them which should create a palindrome. If this is possible in any way, return **"YES"** else return **"NO"**.

Input Format

- The first line contains the string **name**.

Constraints

- **2<=len(stn)<=100**

Output Format

- Return **'YES'** if possible else return **'NO'**.

Evaluation Parameters

- **Sample Input**

```
maria
mira
```

- **Sample Output**

```
YES
```

- **Explanation**

If you remove the substring, 'ari' from 'maria' and 'ira' from 'mira' then the final string after concatenation will be 'ariira'. This name is a palindrome so we return 'YES'.

Solution

PARTIALLY ACCEPTED | SCORE: **61.1** / 100

Code Quality Analysis

Many quality violations
Quality score: 2.2

Deep Code Analysis Results

Straightforward approach
No cyclomatic constructs detected.

Low modularity
Some reusable components found.

Low extensibility
Some extensible features detected.

```java
1  import java.io.*;
2  import java.math.*;
3  import java.security.*;
4  import java.text.*;
5  import java.util.*;
6  import java.util.concurrent.*;
7  import java.util.function.*;
8  import java.util.regex.*;
9  import java.util.stream.*;
10 import static java.util.stream.Collectors.joining;
11 import static java.util.stream.Collectors.toList;
12
13
14 class Result {
15
16         /*
17          * Complete the 'palindromicString' function below.
18          *
19          * The function is expected to return an STRING.
20          * The function accepts following parameters:
21          * 1. STRING father
22          * 2. STRING mother
23          */
24
25
26         public static String palindromicString(String father, String mother) {
27                 // Write your code here
28 //              for(int i=0;i<father.length();i++)
29 //              {
30 //                  for(int j=0;j<mother.length();j++)
31 //                  {
32 //                      if(father.substring(i).equals(new
StringBuilder(mother.substring(j)).reverse().toString()))
33 //                      {
34 //                          return "YES";
35 //                      }
36 //                  }
37 //              }
38 //              return "NO";
39
40         int flen=father.length();
41         int mlen=mother.length();
42         for(int len=1;len<=Math.min(flen,mlen);len++)
43         {
44             for(int i=0;i<=flen-len;i++)
45             {
46                 // for(int j=0;j<=mlen-len;j++)
47                 for(int j=0;j<=Math.min(mlen-1,i+flen-len);j++)
48                 {
49                     if(father.substring(i,i+len).equals(new
StringBuilder(mother.substring(j,j+len)).reverse().toString()))
50                     {
51                         return "YES";
52                     }
53                 }
```

Java 8

```
54                }
55            }
56        return "NO";
57
58
59        }
60 }
61
62 public class Source {
63        public static void main(String[] args) throws IOException {
64                BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(System.in));
65                String father = bufferedReader.readLine();
66                String mother = bufferedReader.readLine();
67
68                String result = Result.palindromicString(father, mother);
69
70                System.out.print(result);
71
72                bufferedReader.close();          }
73 }
```

Evaluation Details

Testcase #8 *(weight:5)*

Status	Failed
Execution time	0.33s
CPU	0s
Memory	2MB
Description	Runtime error.

Solution output

```
Exception in thread "main" java.lang.StringIndexOutOfBoundsException: String index out of
range: 144
at java.lang.String.substring(String.java:1963)
at Result.palindromicString(Source.java:49)
at Source.main(Source.java:68)
```

Testcase #7 *(weight:5)*

Status	Passed
Execution time	0.39s
CPU	0s
Memory	2MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #2 *(weight:1)*

Status	Failed

Execution time	0.41s
CPU	0s
Memory	2MB
Description	Runtime error.

Solution output

```
Exception in thread "main" java.lang.StringIndexOutOfBoundsException: String index out of range: 7
at java.lang.String.substring(String.java:1963)
at Result.palindromicString(Source.java:49)
at Source.main(Source.java:68)
```

Testcase #1 *(sample)*

Status	Passed
Execution time	0.37s
CPU	0s
Memory	2MB
Description	Testcase passed! The solution's output matches the expected output.

Input

```
maria
mira
```

Solution output

YES

Expected output

YES

Testcase #6 *(weight:4)*

Status	Passed
Execution time	0.35s
CPU	0s
Memory	2MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #5 *(weight:1)*

Status	Failed
Execution time	0.36s

CPU	0s
Memory	2MB
Description	Runtime error.

Solution output

```
Exception in thread "main" java.lang.StringIndexOutOfBoundsException: String index out of
range: 42
at java.lang.String.substring(String.java:1963)
at Result.palindromicString(Source.java:49)
at Source.main(Source.java:68)
```

Testcase #4 *(weight:1)*

Status	Passed
Execution time	0.40s
CPU	0s
Memory	2MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #3 *(weight:1)*

Status	Passed
Execution time	0.33s
CPU	0s
Memory	2MB
Description	Testcase passed! The solution's output matches the expected output.

Section 3 : Capgemini-Coding

Problem 1 : Second Largest

CODING | SCORE: **100**

Problem Statement

Given an array of **N** positive integers, find the **2nd** largest integer from the array. **N** will always be greater than or equal to **2**.

Return the **2nd** largest element of the array.

Input Format

- First-line contains the integer **N**.

- Next **N** lines contain ith element of **arr**.

Constraints

- $2<=N<=10^4$

- $1<=arr[i]<=10^4$

Output Format

- Return the **2nd** largest integer of the array.

Evaluation Parameters

- **Sample Input**

```
4
6
322
72
12
```

- **Sample Output**

```
72
```

- **Explanation**

If the array is sorted in descending order, the resulting array will be [322,72,12,6] where 322 will be the largest integer and 72 will be the 2nd largest integer of the group. Therefore, 72 is the answer.

Solution

PARTIALLY ACCEPTED | SCORE: **82.1** / 100

Code Quality Analysis

Many quality violations
Quality score: 2.4

Deep Code Analysis Results

Straightforward approach
No cyclomatic constructs detected.

Very low modularity
No reusable components found.

Low extensibility
Some extensible features detected.

```java
1   import java.io.*;
2   import java.math.*;
3   import java.security.*;
4   import java.text.*;
5   import java.util.*;
6   import java.util.concurrent.*;
7   import java.util.function.*;
8   import java.util.regex.*;
9   import java.util.stream.*;
10  import static java.util.stream.Collectors.joining;
11  import static java.util.stream.Collectors.toList;
12
13
14  class Result {
15
16          /*
17           * Complete the 'secondLargest' function below.
18           *
19           * The function is expected to return an INTEGER.
20           * The function accepts INTEGER ARRAY arr as parameter.
21           *
22           */
23
24
25          public static int secondLargest(List<Integer> arr) {
26                  // Write your code here
27                  int largest=Integer.MIN_VALUE;
28                  int secondLargest=Integer.MIN_VALUE;
29                  for(int i=0;i<arr.size();i++)
30                  {
31                      int curr=arr.get(i);
32                      if(curr>largest)
33                      {
34                          secondLargest=largest;
35                          largest=curr;
36                      }
37                      else if(curr>secondLargest&&curr<largest)
38                      {
39                          secondLargest=curr;
40                      }
41                  }
42                  return secondLargest;
43
44
45          }
46  }
```

Java 8

Powered by doselect

```
47
48  public class Source {
49         public static void main(String[] args) throws IOException {
50                 BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(System.in));
51                 int arrCount = Integer.parseInt(bufferedReader.readLine().trim());
52
53                 List<Integer> arr = IntStream.range(0, arrCount).mapToObj(i -> {
54                     try {
55                             return bufferedReader.readLine().replaceAll("\\s+$", "");
56                     } catch (IOException ex) {
57                             throw new RuntimeException(ex);
58                     }
59             })
60                     .map(String::trim)
61                     .map(Integer::parseInt)
62                     .collect(toList());
63
64                 int result = Result.secondLargest(arr);
65
66                 System.out.print(String.valueOf(result));
67
68                 bufferedReader.close();         }
69  }
```

Evaluation Details

Testcase #2 *(weight:1)*

Status	Passed
Execution time	0.54s
CPU	0s
Memory	30MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #7 *(weight:5)*

Status	Passed
Execution time	0.68s
CPU	0s
Memory	31MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #8 *(weight:5)*

Status	Failed
Execution time	0.77s
CPU	0s
Memory	32MB
Description	Testcase failed! The solution's output doesn't match the expected output.

Testcase #3 *(weight:2)*

Status	Passed
Execution time	0.63s
CPU	0s
Memory	30MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #4 *(weight:3)*

Status	Passed
Execution time	0.65s
CPU	0s
Memory	29MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #5 *(weight:3)*

Status	Passed
Execution time	0.75s
CPU	0s
Memory	32MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #6 *(weight:4)*

Status	Passed
Execution time	1.09s
CPU	0s
Memory	32MB
Description	Testcase passed! The solution's output matches the expected output.

Testcase #1 *(sample)*

Status	Passed
Execution time	0.44s
CPU	0s
Memory	25MB
Description	Testcase passed! The solution's output matches the expected output.

Input

4
6
322
72
12

Solution output

72

Expected output

72

Testcase #9 *(weight:5)*

Status	Passed
Execution time	0.78s
CPU	0s
Memory	31MB
Description	Testcase passed! The solution's output matches the expected output.

Section 4 : Java Full Stack-MCQ

Problem 1 : Element is

What is the **output** of the following code?

```
import java.util.*;
class TestA{
public static void main(String args[]){
ArrayList<String> list=new ArrayList<String>();
list.add("red");
list.add("blue");

String s=list.get(1);
System.out.println("element is: "+s);

Iterator<String> itr=list.iterator();
while(itr.hasNext()){
System.out.println(itr.next());
}
}
}
```

Solution

ACCEPTED SCORE: **1.0** / 1

```
red
blue
element is: blue
```

CANDIDATE'S ANSWER	CORRECT ANSWER
element is: blue red blue	

```
red
blue
```

```
blue
red
```

Problem 2 : Is Not Valid

Which **loop declaration** is not valid?

Solution

CANDIDATE'S ANSWER   **CORRECT ANSWER**

```
for ( int i = 99; i >= 0; i / 9 )
```

```
for ( int i = 7; i <= 77; i += 7 )
```

```
for ( int i = 20; i >= 2; - -i )
```

```
for ( int i = 2; i <= 20; i = 2* i )
```

Problem 3 : Talend: Java Mode

A Java programmer was working with a large amount of data using Talend Studio. He wanted to run the code in Java debug mode. He opens the Run view of **Talend Studio** and finds four options as follows.

Which option does the programmer click to **run the code in Java debug mode**?

Solution

ACCEPTED | SCORE: **1.0** / 1

Basic run

CANDIDATE'S ANSWER   **CORRECT ANSWER**
Debug Run

Memory run

Target Exec

Problem 4 : Do We Use

In order to add a **non-abstract method** to an **interface,** which **keyword** do we use?

Solution

ACCEPTED | SCORE: **1.0** / 1

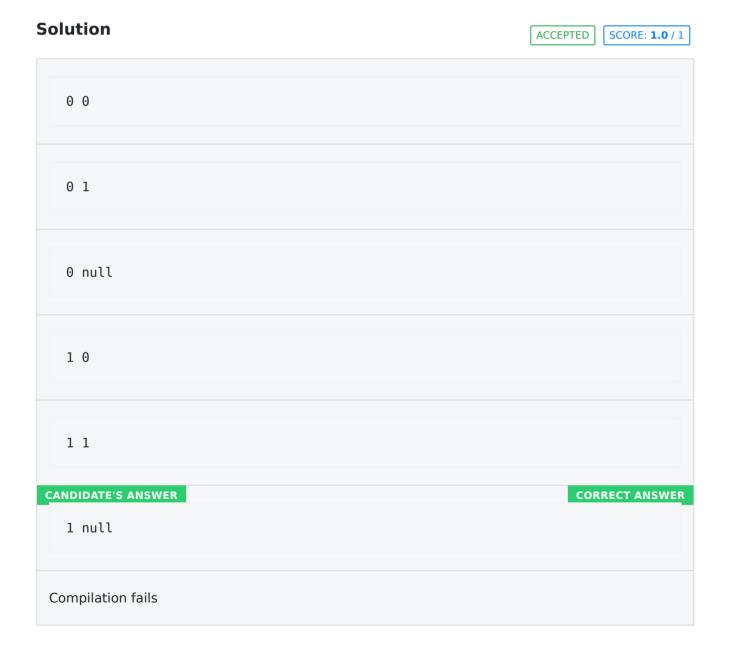| CANDIDATE'S ANSWER	CORRECT ANSWER
Default	
Abstract	
Static	
Abstract Method	

Problem 5 : Catch (exception E)

Given:

```java
public class Test {
    int i1;
    Integer i2;
    public static void main(String[] args) {
        Test test = new Test();
        int sum = 0;
        try {
            sum = test.i1++ + test.i2++;
        } catch (Exception e) { }
        System.out.println(test.i1 + " " + test.i2);
    }
}
```

What is the program's output?

Solution

ACCEPTED | SCORE: **1.0** / 1

0 0

0 1

0 null

1 0

1 1

CANDIDATE'S ANSWER　　　　　　　　　　　　　**CORRECT ANSWER**

1 null

Compilation fails

Problem 6 : Basic Input Keywords

In the abstract class below the author has used some basic input keywords. Find out the **output** of the code below:

```
class A {
public int i;
public int j;
A() {
i = 11;
j = 22;
}
} class B extends A {
int a;
B() {
super();
} } class super {
public static void main(String args[])
{
B obj = new B();
System.out.println(obj.i + " " + obj.j) }
}
```

Solution

22 11

compile time error

Runtime error

CANDIDATE'S ANSWER **CORRECT ANSWER**

11 22

Problem 7 : Running This Code

What will be the result of **running this code**?

```
interface Inter1 {
default void f() {
System.out.println("Inter1");
}
}
interface Inter2 extends Inter1 {
default void f() {
System.out.println("Inter2");
}
}
interface Inter3 {
public static void f() {
System.out.println("Inter3");
}
}
public class InterTest implements Inter2, Inter3 {
public static void main(String[] args) {
InterTest obj= new InterTest();
obj.f();
}
}
```

Solution

ACCEPTED | SCORE: **1.0** / 1

Compiler error because default methods cannot be overridden.

Compiler error because two interfaces cannot be implemented by a class.

Prints "Inter1"

CANDIDATE'S ANSWER   **CORRECT ANSWER**
Prints "Inter2"

Prints "Inter3"

Throws exception at runtime.

None of these

Problem 8 : Run The Thread

Which of the following **methods** will you replace with '**X**' in order to run the **thread**?

```
new Thread(
() -> System.out.println("Hi !");
).X;
```

Solution

ACCEPTED | SCORE: **1.0** / 1

CANDIDATE'S ANSWER	CORRECT ANSWER

start()

run()

runThread()

startThread()

Problem 9 : Entity subclass

MCQ   SCORE: **1**

Consider a situation when an **entity subclass** is added to an existing hierarchy that uses a different access type.

Which of the following option will be used here?

Solution

ACCEPTED   SCORE: **1.0** / 1

Property Access Mode

CANDIDATE'S ANSWER   **CORRECT ANSWER**
Mixed Access Mode

Field Access Mode

None of these

Problem 10 : Output MyClass

```java
public class MyClass {
public static void main(String args[]) {
int x;
try{
method();
}
catch(ArithmeticException e)
{
System.out.println("Exception ocuured in main");
}
}
static void method()
{
try{
int x=8/___;
}
catch(____ e)
{
System.out.println("Exception occurred not in main");
}
}
}
```

output:- **Exception occurred in main**

Fill the Spaces with **correct options** to get the desired output.

Solution

ACCEPTED   SCORE: **1.0** / 1

1
Exception

0
ArithmeticException

CANDIDATE'S ANSWER   **CORRECT ANSWER**

0
NumberFormatException

1
NumberFormatException

Problem 11 : New class java

What will be the output for the **following code**?

```java
class Test {
    public static void main(String[] args) {
        Class<Integer> cls = new Class<>(5);
        System.out.println(cls.getData());
        Class<String> cls2 = new Class<>("Java");
        System.out.println(cls2.getData());
    }
}

class Class<T> {
    private T data;
    public Class(T data) {
        this.data = data;
    }
    public T getData() {
        return this.data;
    }
}
```

Solution

ACCEPTED   SCORE: **1.0** / 1

Java

CANDIDATE'S ANSWER	CORRECT ANSWER
5 Java	

compile error

Run time error

Problem 12 : Topic On Operators

Samuel started a new topic on **operators**. He curiously wrote down a few lines of code. Find the **output:**

```
class operators {
public static void main(String args[])
{
int var1 = 5; int var2 = 14;
int var3;
var3 = ++ var2 * var1 / var2 + var2;
System.out.print(var3);
} }
```

Solution

ACCEPTED SCORE: **1.0** / 1

CANDIDATE'S ANSWER	CORRECT ANSWER
20	
45	
40	
56	

Problem 13 : Find The Output

Find the **output** of the code below:

```
public class String {
public static void main(String[] args) {
StringBuffer s1 = new StringBuffer("Complete");
s1.setCharAt(1,'i');
s1.setCharAt(7,'d');
System.out.println(s1);
} }
```

Solution

ACCEPTED   SCORE: **1.0** / 1

Iomplede

Complete

CANDIDATE'S ANSWER   **CORRECT ANSWER**
Cimpletd

Coipletd

Problem 14 : The Code Below

What will be the output of the below code?

```
class demo{
    public static void main(String[] args){
        String s1 = new String("welcome") ;
        s1.concat("user") ;
        String s2 = s1.concat("class") ;
        s1 = s1.concat("bye") ;
        System.out.println(s1);
    }
}
```

Solution

ACCEPTED | SCORE: **1.0** / 1

welcomeuserbye

userbye

bye

CANDIDATE'S ANSWER   **CORRECT ANSWER**

welcomebye

Problem 15 : The Given Code

Given:

```
List l1 = new ArrayList < > ();
l1.add(1);
List l2 = l1; // Line 1
l2.add(2); // Line 2
System.out.println(l2.size());
```

What happens when **compiling** and **executing** the given code?

Solution

ACCEPTED · SCORE: **1.0** / 1

It prints out a number 1.

It prints out a number 2.

It throws an exception.

Compilation fails on line 1

CANDIDATE'S ANSWER   **CORRECT ANSWER**
Compilation fails on line 2

Problem 16 : Integer Array

What will be the output of the **following code**?

```
public class ClassA{
 public static < E > void printArray(E[] elements) {
 for ( E element : elements){
 System.out.println(element );
 }
 System.out.println();
 }
 public static void main( String args[] ) {
 Integer[] intArray = { 40, 50 };
 Character[] charArray = { 'J', 'A', 'V'};

 System.out.println( "Integer Array" );
 printArray( intArray );

 System.out.println( "Character Array" );
 printArray( charArray );
 }
 }
```

Solution

ACCEPTED   SCORE: **1.0** / 1

CANDIDATE'S ANSWER	CORRECT ANSWER
Integer Array 40 50 Character Array J A V	
Character Array J A V Integer Array 40 50	
Integer Array 40 50	
None of these	

Problem 17 : Int X=0

```java
public class MyClass {
public static void main(String args[]) {
int x=0;
try
{
for(int ele=-1;ele<7;++ele)
{
x+=3/ele;
}
}
catch(Exception e)
{
System.out.println(x++);
}
}
}
```

Understand the above **Code** and **predict the output**.

Solution

1

CANDIDATE'S ANSWER	CORRECT ANSWER
-3	

-2

2

Problem 18 : System Code Segment

The **output** of the following code segment is:

```
public class showarr { public static void main(String[] args) { {
char ar [] = new char[10];
ar[0] = 'a';
for (int i = 0; i < 5; i++) {
System.out.print(ar[i] + " " );
i++;
}
} }
}
```

Solution

ACCEPTED   SCORE: **1.0** / 1

CANDIDATE'S ANSWER	CORRECT ANSWER
a	
a a	
a a a	
a 0 0	

Problem 19 : Class And Constructors

The end-term paper was prepared on the topic of **Class** and **constructors**. The **output** obtained in the code below is of 2 integer values.
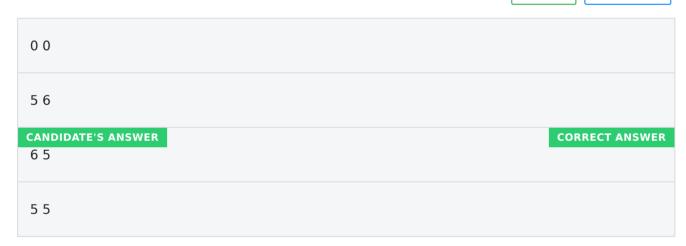
Find out the values:

```
class area {
int width;
int length;
int area;
void area(int width, int length) {
this.width = width;
this.length = length;
}
} class Output {
public static void main(String args[])
{
area obj = new area();
obj.area(5 , 6);
System.out.println(obj.length + " " + obj.width); } }
```

Solution

ACCEPTED   SCORE: **1.0** / 1

0 0

5 6

CANDIDATE'S ANSWER   **CORRECT ANSWER**

6 5

5 5

Problem 20 : Statement(S) Is/Are Correct

Which of the following statement(s) **is/are correct**?

 I. Private members of a class can only be accessed by the members of the same class.

 II. Protected members of a class can be inherited only by a subclass of another package.

III. Protected members of a class can be inherited by a subclass of another package, and become private members of that subclass.

Solution

CANDIDATE'S ANSWER	CORRECT ANSWER
I only	
II only	
I and II only	
I and III only	
None of the statements is true.	