J2EE/TCLD-01M/R12/01
Online Test
Instructor and Examiner:
MD. ABDUL BARI

Test Results

surname	name	user	points
siraj	Sirajuddin Ahmed	siraj	25.567 (57%)

test: R-14 Core Java mock test 3

start time:	2011-12-29 09:30:50
end time:	2011-12-29 10:20:43
time:	00:49:53
test time [min]:	50
basic points:	1.000
points for wrong answer:	0.000
points for no answer:	0.000
max score:	45.000
correct:	25 (56%)

R-14 Core Java mock test 3

#	points	IP	start [hh:mm:ss]	end [hh:mm:ss]	time [mm:ss]	reaction [sec]
1 S	0.000	281473913979009	09:25:50	10:19:51	54:01	60.953

Given:

```
int[][] items =
{ {0, 1, 3, 4},
{4, 3, 99, 0, 7 },
{3, 2} } ;
```

Which of the following fragments prints out every element of items?

	1	for (int row=0; row < items.length; row++)
		{
		System.out.println();
		for (int col=0; col < items[row].length; col++)
		System.out.print(items[row][col] + " ");
		}

	2	for (int row=0; row < items.length; row++)
		{
		System.out.println();
		for (int col=0; col < items[col].length; col++)
		System.out.print(items[row][col] + " ");
		}

-	3	for (int row=0; row < items.length; row++)
		{
		for (int row=0; row < items[row].length; row++)
		System.out.print(items[row][col] + " ");
		System.out.println();
		}

	4	for (int row=0; row < items.length; row++)
		{
		System.out.println();
		for (int col=0; col < items.length; col++)
		System.out.print(items[row][col] + " ");
		}

2 S	0.000	281473913979009	09:27:48	10:20:43	52:55	50.844

Examine the following program fragment:

```
int[] array = { 1, 4, 3, 6, 8, 2, 5};
int what = array[0];

// scan the array
for ( int index=0; index < array.length; index++ )
{
if ( array[ index ] > what )
what = array[ index ];
}
System.out.println( what );
```

What does the fragment write to the monitor?

	1	1
	2	8
	3	5

-	4	1 4 3 6 8 2 5

3 S	1.000	281473913979009	09:30:14	09:34:45	04:31	10.016

What is the base type of the array given below?
Color[] example = new Color[20];

	1	example
+	2	Color
	3	No base type
	4	Both Color and example

4 S	0.000	281473913979009	09:34:45	10:15:59	41:14	32.015

Say that names has been declared

String[] names = new String[10] ;

and that further statements (not shown) have put String references into some of the slots.

Which of the following fragments counts the number of non-null slots in the array?

	1	int count = 0;
		for (int j = 0; j < names.length; j++)
		if (names[j] != null)
		count++ ;

	2	int count = 0;
		while (names[count] != null)
		{
		count++ ;
		}

-	3	int j = 0;
		for (int count = 0; count < names.length; count++)
		if (names[j] != null)
		j++ ;

	4	int j = 0;
		int count = 0;
		while (names[++j] != null)
		count++ ;

5 S	1.000	281473913979009	09:35:10	09:36:24	01:14	73.5

What is the output of the following code fragment:

int[] ar = {2, 4, 6, 8 };

ar[0] = 23;
ar[3] = ar[1];

System.out.println(ar[0] + " " + ar[3]);

	1	2 8
+	2	23 4
	3	23 2
	4	31

6 S	0.000	281473913979009	09:36:24	09:37:10	00:46	45.954

What is the output of the following code fragment:

int[] y = new int[5];

y[0] = 34;
y[1] = 88;

System.out.println(y[0] + " " + y[1] + " " + y[5]);

	1	34 88 88
	2	0 34 88
-	3	34 88 0
	4	The program is defective and will not compile.

7 S	1.000	281473913979009	09:37:10	09:37:41	00:31	31.016

What is the output of the following code fragment:

int[] ar = {2, 4, 6, 8 };
System.out.println(ar[0] + " " + ar[1]);

+	1	2 4
	2	8
	3	6 8
	4	2 6

8 M	0.400	281473913979009	09:37:42	09:37:57	00:15	15.235

Which three statements are true? (Choose three)		
+	1	If a class lacks a no-arg constructor, the compiler always creates a default constructor.
-	2	The default constructor invoked the no-arg constructor of the superclass.
+	3	The default constructor initializes method variables.
-	4	The compiler creates a default constructor only when there are no other constructors for the class.
-	5	The default constructor has the same access as its class.

9 S	0.000	281473913979009	09:37:57	10:16:21	38:24	10.844

Say that names has been declared
String[] names = new String[10] ;
and that further statements (not shown) have put String references into some of the slots.
Which of the following fragments prints out every String, but skips null references?

	1	for (int j = 0; j < names.length && names[j] != null ; j++) System.out.println(names[j]);

	2	for (int j = 0; j < names.length; j++)
		if (names[j] != null)
		System.out.println(names[j]);

-	3	for (int j = 0; names[j] != null; j++) System.out.println(names[j]);
	4	for (int j = 0; j < names.length; j++) System.out.println(names[j]);

10 M	0.500	281473913979009	09:39:09	09:40:25	01:16	75.937

Given:
1. public interface Foo {
2. int k = 4;
3. }
Which three are equivalent to line 2? (Choose three)

-	1	static int k = 4;
-	2	final int k = 4;
+	3	protected int k = 4;
+	4	volatile int k = 4;
+	5	abstract int k = 4;
-	6	public int k = 4;

11 S	1.000	281473913979009	09:40:25	10:16:59	36:34	34.828

Given:
12. float f[][][] = new float[3][][];
13. float f0 = 1.0f;
14. float[][] farray = new float[1][1];
What is valid?

	1	f[0] = farray[0];
	2	f[0] = farray[0][0];
+	3	f[0] = farray;
	4	f[0] = f0;

12 S	1.000	281473913979009	09:41:50	09:42:14	00:24	23.687

An Outer class which has all its constructors declared as private

	1	has to be declared final.
	2	Cannot be extended.
	3	Cannot be instantiated by any other Outer class.
+	4	Cannot be instantiated by any other Outer class and Cannot be extended.

13 M	1.000	281473913979009	09:42:14	09:43:47	01:33	92.562

Which two cause a compiler error? (Choose two)

+	1	float[] = new float(3);
+	2	float f2[] = new float[];
+	3	float[] f1 = new float[3];
+	4	float f4[] = new float[] { 1.0f. 2.0f. 3.0f};
+	5	float f5[] = { 1.0f, 2.0f, 2.0f };
+	6	float f3[] = new float[3];

14 S	1.000	281473913979009	09:43:47	09:44:17	00:30	30.281

Examine the following:

double[][] values =
{ {1.2, 9.0, 3.2},
{9.2, 0.5, 1.5, -1.2},
{7.3, 7.9, 4.8} } ;

what is in values[2][1] ?

	1	9.2
	2	7.3
+	3	7.9

	4	There is no such array element.

15 S	1.000	281473913979009	09:44:17	09:45:37	01:20	79.203	
Which of the following declares an array of int named img?							
+	1	int[] img;					
	2	new int img[];					
	3	int img;					
	4	int img = int[];					

16 S	1.000	281473913979009	09:45:37	09:46:33	00:56	55.187	
What is the output of the following code fragment: int[] egArray = { 2, 4, 6, 8, 10, 1, 3, 5, 7, 9 }; for (int index= 0 ; index < 5 ; index++) System.out.print(egArray[index] + " ");							
+	1	2 4 6 8 10					
	2	2 4 6 8 10 1 3 5 7 9					
	3	2 4 6 8 10 1					
	4	2 4 6 8					

17 S	1.000	281473913979009	09:46:33	09:47:01	00:28	27.36	
What are the legal indexes for the array ar, given the following declaration: int[] ar = {2, 4, 6, 8 }							
	1	2, 4, 6, 8					
	2	0, 2, 4. 6					
	3	1, 2, 3, 4					
+	4	0, 1, 2, 3					

18 S	0.000	281473913979009	09:47:01	09:48:12	01:11	70.469	
What does the following statement do? int[] values = new int[10] ;							
-	1	It declares values to be a reference to an array object, but initializes it to null					
	2	It declares values to be a reference to an array object and constructs an array object containing 10 integers which are initialized to zero.					
	3	It declares values to be a reference to an array object which does not yet exist, but will contain 10 zeros when it does					
	4	It declares values to be a reference to an array which contains 10 references to int variables.					

19 S	1.000	281473913979009	09:48:12	09:48:52	00:40	36.125	
What is the difference between String rats; and String[] rats; ?							
	1	There is no difference; both declare rats to be a reference variable					
	2	The first constructs a single String object; the second constructs an array of String objects.					
	3	The first initializes rats to null; the second initializes rats to an array of nulls					
+	4	The first declares rats to be a reference to a String object, the second declares rats to be a reference to an array of String references					

20 S	1.000	281473913979009	09:48:52	09:49:18	00:26	26.016	
Given the following: double[][] things = { {1.2, 9.0}, {9.2, 0.5, 0.0}, {7.3, 7.9, 1.2, 3.9} } ; What is the value of things.length ?							
	1	9					
+	2	3					
	3	4					
	4	2					

21 S	0.000	281473913979009	09:49:18	09:50:25	01:07	66.015	
What does the following statement do? String glarch;							
	1	It constructs a String object which will contain the characters "glarch" .					
	2	It declares a reference variable glarch which is initialized to null.					
-	3	It declares an array of String objects named glarch.					
	4	It constructs a String object named glarch.					

22 S	1.000	281473913979009	09:50:25	09:51:53	01:28	87.282	
You want to create a table that looks like:							

12 -9 8
7 14
-32 -1 0

Which of the following will work?

+	1	double[][] table = { {12, -9, 8}, {7, 14}, {-32, -1, 0} };
	2	double[][] table = { 12, -9, 8, 7, 14, -32, -1, 0} ;
	3	double[][] table = { {12, -9, 8}, {7, 14, 0}, -32, -1, 0} };
	4	double[][] table = { {12, -9, 8} {7, 14} {-32, -1, 0} };

23 S	0.000	281473913979009	09:51:53	09:53:01	01:08	68.031

Given the following:

long[][] stuff ;

Which of the following statements constructs an array with 5 rows of 7 columns each and assign its reference to stuff ?

-	1	stuff = new stuff[5][7] ;
	2	stuff = long[7][5] ;
	3	stuff = long[5][7] ;
	4	stuff = new long[5][7] ;

24 S	1.000	281473913979009	09:53:01	09:54:57	01:56	115.188

Given:
1. public class ArrayTest {
2. public static void main(String[] args) {
3. float fl[], f2[];
4. fl = new float[10];
5. f2 = f1;
6. System.out.println("f2[0]= " + f2[0]);
7. }
8. }
What is the result?

	1	An error at line 6 causes an expectation at runtime.
	2	An error at line 6 causes compile to fail.
+	3	It prints f2[0] = 0.0.
	4	It prints f2[0] = NaN.
	5	An error at line 5 causes compile to fail.

25 M	0.667	281473913979009	09:54:57	09:56:11	01:14	73.203

Which three form part of correct array declarations? (Choose three)

+	1	private int [3] a []
+	2	static int [] a
-	3	public [] int a
-	4	public final int [] a
+	5	private int a [3]
+	6	public int a []

26 S	1.000	281473913979009	09:56:11	09:56:37	00:26	26.203

What is the meaning of null?

	1	It is another name for zero.
	2	It is the String object that contains no characters.
+	3	A reference variable that contains null is not referring to an object.
	4	It is a special value used to indicate an error condition.

27 S	0.000	281473913979009	09:56:37	09:59:20	02:43	162.984

What is the output of the following code fragment:

int[] egArray = { 2, 4, 6, 8, 10, 1, 3, 5, 7, 9 };

for (int index= 0 ; index < egArray.length ; index = index + 2)
System.out.print(egArray[index] + " ");

	1	2 6 10 3 7

	2	4 8 1 5 9
-	3	2 4 6 8 10 1 3 5 7 9
	4	2 6 10 3 7 0

28 S	1.000	281473913979009	09:59:20	10:00:06	00:46	45.172

Given the following:

```
int[][] items =
{ {0, 1, 3, 4},
{4, 3, 99, 0, 7 },
{3, 2} } ;
```

Which of the following statements replaces the 99 with 77?

	1	items[2][1] = 77;
	2	items[2][3] = 77;
+	3	items[1][2] = 77;
	4	items[99] = 77;

29 S	0.000	281473913979009	10:00:06	10:00:38	00:32	31.844

For which of the following applications is an array NOT suitable:

-	1	Holding the total sales a store made in each of twelve months.
	2	Holding the scores on twelve midterms exams of a class.
	3	Holding the name, social security number, age, and income of one individual.
	4	Holding the temperature readings taken every hour throughout a day.

30 S	0.000	281473913979009	10:00:38	10:00:54	00:16	14.875

You want a class to have access to members of another class in the same package. Which is the most restrictive access that accomplishes this objective?

	1	default access
-	2	private
	3	public
	4	transient
	5	protected

31 S	0.000	281473913979009	10:00:54	10:01:52	00:58	57.938

Does a programmer always know how long an array will be when the program is being written?

-	1	Yes---the program will not compile without the length being declared.
	2	No---arrays can grow to whatever length is needed.
	3	No---the array object is created when the program is running, and the length might change from run to run.
	4	Yes---otherwise the program will not run correctly.

32 S	1.000	281473913979009	10:01:53	10:02:32	00:39	39.485

Given the declaration
String[] names = new String[10] ;
Which of the following statements puts a reference to the String "Hello" in the last slot of the array?

	1	names[0] = "Hello" ;
+	2	names[9] = "Hello" ;
	3	names[10] = "Hello" ;
	4	String[names.length-1] = "Hello" ;

33 S	0.000	281473913979009	10:02:32	10:03:25	00:53	52.516

Given:

```
1. public class Test {
2. public static void main(String [] args) {
3. System.out.println(args.length > 4 &&
4. args[4].equals("-d"));
5. }
6. }
```

If the program is invoked using the command line:
java Test One Two Three –d
What is the result?

	1	false
	2	Compilation fails.
-	3	An exception is thrown at runtime.
	4	true

34 S	1.000	281473913979009	10:03:25	10:04:53	01:28	86.922

Fill in the blank in the following code fragment so that each element of the array is assigned twice the value of its index.

```
int[] array = new int[10];

// scan the array
for ( int index=0; index < array.length; index++ )
```

{
_____
}

+	1	array[index] = 2*index;
	2	array[2*index] = 2*index;
	3	array[index] = 2*array[index];
	4	index = 2*index;

35 S	1.000	281473913979009	10:04:53	10:05:37	00:44	43.593

What does the following statement do?

int[] values = new int[10] ;

	1	It declares values to be a reference to an array which contains 10 references to int variables.
+	2	It declares values to be a reference to an array object and constructs an array object containing 10 integers which are initialized to zero.
	3	It declares values to be a reference to an array object, but initializes it to null.
	4	It declares values to be a reference to an array object which does not yet exist, but will contain 10 zeros when it does.

36 S	1.000	281473913979009	10:05:37	10:06:05	00:28	27.797

Assume the following:
int[] a = {25, -3, 6, 0, -3, 6};
int[] b = new int[100];
int[] c = null;
_____ Value of a.length ?

+	1	6
	2	2
	3	4
	4	3

37 S	0.000	281473913979009	10:06:05	10:06:50	00:45	44.25

What is the output of the following code fragment:

int[] z = new int[9];

z[0] = 7;
z[1] = 3;
z[2] = 4;

System.out.println(z[0] + z[1] + " " + z[5]);

-	1	7 3 0
	2	The program is defective and will not compile.
	3	7 3 4
	4	10 0

38 S	0.000	281473913979009	10:06:50	10:07:26	00:36	36.219

Given:
1. public class Test {
2. private static float[] f = new float[2];
3. public static void main(String args[]) {
4. System.out.println("f[0] = " + f[0]);
5. }
6. }
What is the result?

	1	f[0] = 0.0
-	2	f[0] = 0
	3	An exception is thrown at runtime.
	4	Compilation fails.

39 S	1.000	281473913979009	10:07:26	10:17:26	10:00	21.797

Say that names has been declared

String[] names = new String[10] ;

and that further statements (not shown) have put String references into some of the slots.

Which of the following fragments prints out every String, but skips null references?

	1	for (int j = 0; j < names.length && names[j] != null ; j++) System.out.println(names[j]);
	2	for (int j = 0; j < names.length; j++) System.out.println(names[j]);
	3	for (int j = 0; names[j] != null; j++) System.out.println(names[j]);

+	4	for (int j = 0; j < names.length; j++)
		if (names[j] != null)
		System.out.println(names[j]);

40 S	0.000	281473913979009	10:08:32	10:09:57	01:25	84.782

What is the output of the following code fragment:

```
int[] zip = new int[5];

zip[0] = 7;
zip[1] = 3;
zip[2] = 4;
zip[3] = 1;
zip[4] = 9;

System.out.println( zip[ 2 + 1 ] );
```

-	1	4 3
	2	4
	3	3 7
	4	1

41 S	0.000	281473913979009	10:09:57	10:17:42	07:45	16.563

Say that names has been declared
String[] names = new String[10] ;
and that further statements (not shown) have put String references into some of the slots.
Which of the following fragments prints out the slots of the array from last to first, skipping slots that contain null?

	1	for (int j = names.length-1; j >= 0; j--)
		if (names[j] != null)
		System.out.println(names[j]);

-	2	for (int j = names.length; j >= 0; j++)
		if (names[j] != null)
		System.out.println(names[j]);

	3	for (int j = names.length; j < names.length; j++)
		if (names[j] != null)
		System.out.println(names[j]);

	4	for (int j = 0; j < names.length; j++)
		if (names[j] != null)
		System.out.println(names[j]);

42 S	1.000	281473913979009	10:10:03	10:18:17	08:14	33.859

Fill in the blanks of the following code fragment so that the elements of the array are printed in reverse order, starting with the last element.

```
int[] egArray = { 2, 4, 6, 8, 10, 1, 3, 5, 7, 9 };

for ( int index= _________ ; _______________ ; _________________ )
System.out.print( egArray[ index ] + " " );
```

	1	index = 0; index < egArray.length; index--
	2	index = length-1; index > 0; index--
+	3	index = egArray.length-1; index >= 0; index--
	4	index = length; index < 0; index--

43 S	1.000	281473913979009	10:11:10	10:11:36	00:26	25.547

Given the following:

```
double[][] things =
{ {1.2, 9.0},
{9.2, 0.5, 0.0},
{7.3, 7.9, 1.2, 3.9} } ;
```

What is the value of things[2].length ?

+	1	4
	2	9
	3	3
	4	2

44 S	1.000	281473913979009	10:11:36	10:12:41	01:05	64.375

You want to limit access to a method of a public class to members of the same class.
Which access accomplishes this objective?

	1	default access
	2	transient
	3	protected
	4	public
+	5	private

45 S	0.000	281473913979009	10:12:41	10:13:35	00:54	52.859

What is the output of the following code fragment:

```
int[] zip = new int[5];
```

```
zip[0] = 7;
zip[1] = 3;
zip[2] = 4;
zip[3] = 1;
zip[4] = 9;

int j = 3;

System.out.println( zip[ j-1 ] );
```

	1	1
	2	7
-	3	3
	4	4

topics

points	correct points	module correct	module topic
25.567 / 45 (57%)	**25 / 45 (56%)**	**Core Java**	
	15 / 27 (56%)	15 / 27 (56%)	Array (new) 4-1
	4 / 7 (57%)	4 / 7 (57%)	Array 4-1
	0.4 / 1 (40%)	0 / 1 (0%)	Constructor 5-3
	0.5 / 1 (50%)	0 / 1 (0%)	Access Modifier 6-3
	1 / 2 (50%)	1 / 2 (50%)	General2 4-1
	1 / 1 (100%)	1 / 1 (100%)	Constructor 4-1
	1 / 1 (100%)	1 / 1 (100%)	Array 6-2
	2 / 2 (100%)	2 / 2 (100%)	General2 5-1
	0.667 / 1 (67%)	1 / 1 (100%)	Array 6-3
	0 / 1 (0%)	0 / 1 (0%)	Access Modifier 5-1
	0 / 1 (0%)	0 / 1 (0%)	Access Modifier 4-1