Note: Correct responses are based on Java, J2sdk v 5.0, from Sun Microsystems, Inc. All provided code segments are intended to be syntactically correct, unless otherwise stated (i.e. error is an answer choice) and any necessary Java 2 Standard Packages have been imported. Ignore any typographical errors and assume any undefined variables are defined as used.

| QUESTION 1 | 248 - 178 = ? |
| A. 1012 | B. 910 | C. 196 | D. A16 | E. 10102 |

| QUESTION 2 | What is output by the code to the right? | byte fun = 'A';
out.println(fun); |
| A. 0 | B. 65 | C. 97 | D. A | E. error |

| QUESTION 3 | What is output by the code to the right? | double c = 2;
c *= 5 + 3;
out.println(c); |
| A. 16 | B. 13.0 | C. 11.0 | D. 16.0 | E. 13 |

| QUESTION 4 | What is output by the code to the right? | double d = Math.max(8.3,5.8);
d = Math.round(d);
out.println(d); |
| A. 8.0 | B. 6.0 | C. 8 | D. 6 | E. 5 |

| QUESTION 5 | What is output by the code to the right? | int e = (((11+5)/5) - (3*-2));
out.println(e); |
| A. -1 | B. 3 | C. -6 | D. 8 | E. 9 |

| QUESTION 6 | What is output by the code to the right? | String f = "valentimesday";
out.println(f.indexOf("m")); |
| A. -1 | B. 4 | C. 7 | D. 3 | E. 6 |

| QUESTION 7 | What is output by the code to the right? | String g = "yo";
String h = "go";
out.println(g.length()==h.length()); |
| A. 0 | B. 1 | C. true | D. false | E. -1 |

| QUESTION 8 | What is output by the code to the right? | int i = 9;
i = i%20;
out.println(i); |
| A. 20 | B. 11 | C. 0 | D. 9 | E. -5 |

| QUESTION 9 | What is output by the code to the right? | int j = 19;
do
  j -= 4;
while(j>1);
out.println(j); |
| A. 3 | B. 7 | C. 1 | D. -1 | E. 0 |

| QUESTION 10 | What is output by the code to the right? | int[] k = new int[5];
int[] m = new int[5];
System.arraycopy(m,0,k,0,k.length);
k[2] = 6;
out.println(k[2] + m[2]); |
| A. 6 | B. 0 | C. 12 | D. 5 | E. 18 |

| QUESTION 11 | What is the last value output by the code to the right? | for(double p=0; p<12.95; p+=2.5)
out.println(p); |
| A. 5.0 | B. 7.5 | C. 10.0 | D. 12.5 | E. 15.0 |
### QUESTION 12
What is output by the code to the right?

<table>
<thead>
<tr>
<th>A. 8.173</th>
<th>B. 8.2</th>
<th>C. 8.17</th>
<th>D. 8.174</th>
<th>E. 8.1</th>
</tr>
</thead>
</table>

```
double n = 8.1735f;
String o = String.format("%.3f", n);
out.println(o);
```

### QUESTION 13
What is output by the code to the right?

<table>
<thead>
<tr>
<th>A. 12</th>
<th>B. 20</th>
<th>C. 17</th>
<th>D. 19</th>
<th>E. 23</th>
</tr>
</thead>
</table>

```
int q = 17;
int r = 23;
out.println(q & r ^ 9 & q >> 3);
```

### QUESTION 14
What correctly replaces `<*1>` in the code to the right?

<table>
<thead>
<tr>
<th>A. TreeSet</th>
<th>B. HashSet</th>
<th>C. Set</th>
</tr>
</thead>
<tbody>
<tr>
<td>D. A and B only</td>
<td>E. A, B, and C</td>
<td></td>
</tr>
</tbody>
</table>

```
Set setTest;
setTest = new `<*1>` <Double>();
```

### QUESTION 15
What is output by the code to the right?

<table>
<thead>
<tr>
<th>A. 0</th>
<th>B. 1</th>
<th>C. false</th>
<th>D. true</th>
<th>E. n</th>
</tr>
</thead>
</table>

```
boolean s=true;
boolean t=false;
out.println((!(s&&t)||s ||(t||s)));
```

### QUESTION 16
What replaces `<*1>` in the code to the right so method doEet will return true only when parameter x is equal to the remainder of x divided by parameter y?

| A. return x / y == x; |
| B. return x / y == y; |
| C. return x % y == x; |
| D. return y % y == x; |
| E. more than one of these |

```
public boolean doEet(double x, int y) {
    `<*1>`
}
```

### QUESTION 17
What is output by the code to the right?

<table>
<thead>
<tr>
<th>A. 24</th>
<th>B. 25</th>
<th>C. 17</th>
<th>D. 18</th>
<th>E. error</th>
</tr>
</thead>
</table>

```
Integer[] u = {3,4,5,6,7};
int theSum = 0;
for(int v : u) {
    theSum += v;
}
out.println(theSum);
```

### QUESTION 18
What correctly replaces `<*1>` in the code to the right so all elements of chop are inspected?

| A. chop.nextInt() |
| B. chop.next() |
| C. chop.hasNext() |
| D. chop.hasNextInt() |
| E. chop |

```
String toChop = "1+2/*3+4/5";
Scanner chop = new Scanner(toChop);
chop.useDelimiter("[+/*]");
String output = "";
while( `<*1>` ) {
    output = output + chop.next();
}
out.println(output);
```

### QUESTION 19
Assume `<*1>` was replaced correctly. What is output by the code to the right?

<table>
<thead>
<tr>
<th>A. 12345</th>
<th>B. 15</th>
<th>C. 345</th>
</tr>
</thead>
<tbody>
<tr>
<td>D. 1+2+/*3+4/5</td>
<td>E. 1+23+4/5</td>
<td></td>
</tr>
</tbody>
</table>

### QUESTION 20
Which of the following is a subinterface of Map?

| A. TreeMap | B. TreeSet | C. BinaryMap | D. SortedMap | E. HashSet |
**QUESTION 21**

Which of the following is not a descendant of List?

<table>
<thead>
<tr>
<th>A. Stack</th>
<th>B. ArrayList</th>
<th>C. Queue</th>
<th>D. LinkedList</th>
<th>E. Vector</th>
</tr>
</thead>
</table>

**QUESTION 22**

What is output by the code to the right?

<table>
<thead>
<tr>
<th>A. -1</th>
<th>B. 1</th>
<th>C. 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>D. what</td>
<td>E. error</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 23**

What is returned by the call `dude(3)`?

<table>
<thead>
<tr>
<th>A. -27</th>
<th>B. -35.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. -22.0</td>
<td>D. -27.0</td>
</tr>
<tr>
<td>E. runtime error</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 24**

What is returned by the call `dude(-6)`?

<table>
<thead>
<tr>
<th>A. 2.0</th>
<th>B. 1.0</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. -1.0</td>
<td>D. -2.0</td>
</tr>
<tr>
<td>E. runtime error</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 25**

What is the output of `// line 1`?

<table>
<thead>
<tr>
<th>A. [1,1,1]</th>
<th>B. [0,0,0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. [2,4,6]</td>
<td>D. [1,2,3]</td>
</tr>
<tr>
<td>E. [3,3,3]</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 26**

What is the output of `// line 2`?

<table>
<thead>
<tr>
<th>A. [1,1,1]</th>
<th>B. [0,0,0]</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. [2,4,6]</td>
<td>D. [1,2,3]</td>
</tr>
<tr>
<td>E. [3,3,3]</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 27**

Which of the following could fill `<*1>`?

<table>
<thead>
<tr>
<th>A. new QBack();</th>
<th>B. new RunningBack();</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. new Back();</td>
<td>D. A and B only</td>
</tr>
<tr>
<td>E. A, B, and C</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 28**

Which of the following could fill `<*2>`?

<table>
<thead>
<tr>
<th>A. new QBack();</th>
<th>B. new RunningBack();</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. new Back();</td>
<td>D. A and B only</td>
</tr>
<tr>
<td>E. A, B, and C</td>
<td></td>
</tr>
</tbody>
</table>

**QUESTION 29**

Which of the following could fill `<*3>`?

<table>
<thead>
<tr>
<th>A. new QBack();</th>
<th>B. new RunningBack();</th>
</tr>
</thead>
<tbody>
<tr>
<td>C. new Back();</td>
<td>D. A and B only</td>
</tr>
<tr>
<td>E. A, B, and C</td>
<td></td>
</tr>
</tbody>
</table>
**QUESTION 30**

Assuming a binary search tree has 5 leaves, what is the minimum number of levels that it could have?

A. 0  B. 1  C. 2  D. 3  E. 4

**QUESTION 31**

Which of the following methods is called when adding a new pair to a Map?

A. add()  B. set()  C. insert()  D. put()  E. more than one of these

**QUESTION 32**

What replaces `<*1>` in the code to the right to correctly complete method `What`?

A. `chop.nextInt();`  B. `chop.nextIntInt();`  C. `chop.nextDouble();`  D. A and B only  E. A, B, and C

**QUESTION 33**

What is method `isWhat` trying to determine about `num`?

A. `is` num is a binary number  B. if num is smaller than all numbers in stuff  C. if num is larger than all numbers in stuff  D. if num is not present in stuff  E. if num is present in stuff

**QUESTION 34**

Assume `<*1>` was filled correctly. What is the output of `//line 1`?

A. 0  B. 1  C. true  D. false  E. syntax error

**QUESTION 35**

Assume `<*1>` was filled correctly. What is the output of `//line 2`?

A. 0  B. 1  C. true  D. false  E. syntax error

class What
{
    private int[] stuff;
    public What(String s)
    {
        Scanner chop = new Scanner(s);
        stuff = new int[chop.nextInt()];
        /*<*1>*/
        for(int i=0; i<stuff.length;i++)
            stuff[i]=chop.nextInt();
    }
    public boolean isWhat(int num)
    {
        for(int i=0;i<stuff.length;i++)
        {
            if(num<=stuff[i])
                return false;
        }
        return true;
    }
}

//test code in the main of another class
What runner = new What("5 1 3 4 5 6");
out.println(runner.isWhat(7));  //line 1
out.println(runner.isWhat(6));  //line 2
**QUESTION 36**
What replaces `<*1>` in the code to the right so the constructor in `Bling` would be complete?

A. super();
B. super(costPerLink, size);
C. super(cost);
D. super(cost, 0.0);
E. more than one of these

**QUESTION 37**
What replaces `<*2>` in the code to the right so the `Bling` `getValue()` method would be complete?

A. super.getValue(cost);
B. getValue();
C. super.getValue();
D. A and B
E. A, B, and C

**QUESTION 38**
What replaces `<*3>` in the code to the right so that the `Bling` `toString()` method would be complete?

A. toString();
B. super.toString();
C. chain + " " + costPerLink + " "
D. A and B
E. A, B, and C

**QUESTION 39**
What replaces `<*4>` in the code to right so that reference `shiny` would refer to a `Bling`?

A. new Bling('A',2);
B. new Bling(5.3,7);
C. new Bling(9.9,1.1);
D. new Bling(Math.PI,22);
E. more than one of these

**QUESTION 40**
Assume all blanks have been filled correctly. After executing the test code, which of the following could have been output by `//line 1`?

A. 0  B. 234324.11  C. false  D. bling  E. 400

```java
class Link {
    //code not shown
}
class Chain {
    private ArrayList<Link> chain;
    private double costPerLink;
    public Chain(double cost) {
        //code not shown
    }
    public Chain(double cost, int length) {
        //code not shown
    }
    public double getValue() {
        //code not shown
    }
    public String toString() {
        //code not shown
    }
}
class Bling extends Chain {
    private double blingPriceUp;
    public Bling(double cost, double bling) {
        blingPriceUp = bling;
    }
    public double getValue() {
        return blingPriceUp * `<*2>`
    }
    public String toString() {
        return `<*3>` + " " + blingPriceUp;
    }
}
//test code in the main of another class
Bling shiny = `<*4>`;
out.println(shiny.getValue());  //line 1
```