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

$10011_2 - 101_2 = ?$

A. 1010_2 B. 10000_2 C. 10101_2 D. 11101_2 E. 1110_2

QUESTION 2

What is the output?

A. 0.25 B. 2 C. 1 D. 0 E. 0.5

```
int x = 4, y = 9, z = 29;
out.println(z % y / x);
```

QUESTION 3

What is the output?

A. 10 5 B. 10 4
C. 16 4 D. 11 5 E. 16 5

```
int i, x = 0;
for(i=0; i<5; i++)
x += i;
x++;
out.println(x + " " + i);
```

QUESTION 4

What is the output if the input is shown to the right?

```
1 2 3 4
5 6 7 8
```

A. 21 B. 20
C. 36 D. 15 E. a run-time error will occur

```
Scanner kb = new Scanner(in);
int x = kb.nextInt();
for(int i=0; i<5; i++)
  x += kb.nextInt();
out.println(x);
```

QUESTION 5

What is the output?

A. 1 B. -2 C. -1 D. 0 E. 2

QUESTION 6

How many times will this `while` loop iterate?

A. 15 B. 14 C. 6 D. 7 E. 8

```
int r = 7;
while(r >= 0) {
  r += 2;
  r -= 3;
}
out.println(r);
```

QUESTION 7

What is the output?

A. 6 6 B. 8 8
C. 7 7 D. 6 8 E. infinite loop

```
int x = 4, y = 10;
do {
  x++;
  y--;
} while(x != y);
out.println(x + " " + y);
```

QUESTION 8

What replaces **<*1>** so all the elements of `a` will be inspected?

A. `a.size()` B. `a.length()`
C. `a.size` D. `a.length` E. `a.length()-1`

QUESTION 9

Assume **<*1>** was filled correctly. What is the output?

A. 5 B. 4 C. 6 D. 7 E. 8

```
int[] a = {3,4,5,6,7};
int x = <*1> ;
for(int i=0; i<x; i++) {
  // code not shown
}
out.println(x);
```

QUESTION 10

What is the output?

A. CATE B. NCATE
C. CATEN D. CAT E. NCA

```
String s = "CONCATENATION";
out.println(s.substring(2,5));
```

QUESTION 11

What is the output?

A. `125` B. `123456`

C. `246` D. `1256` E. runtime error

```
int[] a = {1,2,3,4,5,6};
for(int i : a) {
  out.print(i);
  i++;
}
```

QUESTION 12

How many `*`'s will be printed by this code segment if `n = 10`?

A. 36 B. 90 C. 50 D. 45 E. 100

QUESTION13

What is the expected running time of this code segment? Choose the smallest correct value.

A. `O(1)` B. `O(n)`

C. `O(n)` D. $O(n \log_2 n)$ E. $O(n^2)$

```
int n;
for(int i = 1; i < n; i++)
 for(int j = 1; j < n; j+=2)
   out.print("*");
```

QUESTION 14

What is returned by `fun(10)`?

A. `10` B. `9`

C. `5` D. `0` E. infinite recursion

```
public static int fun(int x) {
 if(x != 0)
   return fun(x-2) + 1;
 else
   return x;
}
```

QUESTION 15

What is the output?

A. `14.12006` B. `2017.1`

C. `11.22006` D. `74.12006` E. `722.12006`

```
int x = 7;
double y = 2.1;
String s = "2006";
out.print(x + 2 * y + s);
```

QUESTION 16

What is the output?

A. `1` B. `16` C. `17` D. `41` E. `408`

```
out.println(17 & 24);
```

QUESTION 17

What is the output?

A. `COLE` B. `LEGE`

C. `LLEG` D. `COL` E. runtime error

```
StringBuffer sb = new
          StringBuffer("COLLEGE");
sb.delete(3,6);
out.print(sb);
```

QUESTION 18

What is the output?

A. `true true true` B. `true true false`

C. `true false false` D. `true false true`

E. `false false false`

```
String s = new String("Texas");
String t = "Texas";
String u = "TEXAS";
u = t;
out.print(s.equals(t) + " ");
out.print(s == t);
out.print(" " + t.equals(u));
```

QUESTION 19

How many lines will be printed by this code segment?

A. 6 B. 7 C. 8 D. 9 E. 5

QUESTION 20

What are the ***first*** and ***last*** values output?

A. `75 1` B. `37 1`

C. `75 0` D. `37 0` E. `75 -1`

```
int i = 75;
do {
 out.println(i);
}while((i /= 2) > 0);
```

QUESTION 21

What replaces **<*1>** so the output from the code in **line 2** is:

JAVA ROCKS

A. `this();` B. `out.print("JAVA ");`

C. either A or B D. **<*1>** should be left blank

E. this output is impossible

QUESTION 22

What is the output of **line 3**?

A. 1 B. 2

C. 3 D. 4 E. 0

QUESTION 23

What is the output of **line 4**?

A. 0 B. 6

C. 7 D. 3 E. 4

QUESTION 24

The variable `num` is best described as a(n):

A. private variable B. instance variable

C. free variable D. object variable

E. class variable

```
class A extends Count {
  A() {
    num++;
    out.print("JAVA ");
  }
}
class B extends A {
  B() {
    // <*1>
    num++;
    out.print("ROCKS ");
  }
}
public class Count {
  static int num = 0;
  public void create() {
    A a = new A();

    B b = new B();              // line 2
  }
  public static void main(String args[])
  {
    Count c = new Count();
    Count co = new Count();
    c.create();
    out.println(num);           // line 3
    co.create();
    out.println(num);           // line 4
  }
}
```

QUESTION 25

What is the output?

A. 4 4 3 B. 3 4 3

C. 4 3 3 D. 4 4 4 E. 3 4 4

QUESTION 26

What is the value of `u[1]` after this code segment is executed?

A. `my.name@yah` B. `o` C. `m` D. `.c`

E. an empty string

```
String a = "my.name@yahoo.com";
String[] s = a.split("m");
String[] t = a.split("o");
String[] u = a.split("o+");
out.println(s.length + " " +
              t.length + " " + u.length);
```

QUESTION 27

What is the output if **< ** >** are replaced with `String`?

A. `*-*-/` B. `*///`

C. `*-/*-/` D. `*/-/*/-/`

E. run time error

QUESTION 28

What is the output if **< ** >** are replaced with `Integer`?

A. `-*-*/` B. `-/*/-/*/`

C. `-/*-/-/` D. `-/*/-/-/`

E. run time error

```
Object x[] = new Object[] { "Java",
                    new Integer(7),
                    new String("Rocks"),
                    new Character('!')};
for (int i=0;i<x.length;i++)
  try {
    < ** >  a = new < ** >((< ** >) x[i]);
    out.print("*");
  }
  catch (Exception e) {
    out.print("-");
  }
  finally {
     out.print("/");
}
```

QUESTION 29

Which of these returns `true`?

A. `check(60)` B. `check(18)`

C. `check(-36)` D. `check(65)`

E. more than one of these

QUESTION 30

Which of these describes the behavior of `check()`?

A. Checks whether `n` is divisible by `12`

B. Checks whether `n` is even

C. Checks whether `n` is a base `3` number

D. Checks whether `n` is positive

E. none of these because `case 2` is not considered

```
public static boolean check(int n)
{
  switch(n%3)                        {
    case 0: return (n%4 == 0);
    case 1: return false;
    default: return false;
  }
}
```

QUESTION 31

What is the output?

A. `9` B. `0` C. `8` D. `4` E. `-3`

```
int[] a = {12, 4, -3, 8, 9, 0};
Arrays.sort(a);
out.print(a[2]);
```

QUESTION 32

What is the output?

A. `raindrop` B. `raindrops`

C. `rain` D. `raindroprain`

E. syntax error

```
String s = "rain" + "drop" + 's';
System.out.print(s);
```

QUESTION 33

Which of the following replaces **`< *1 >`** so this code will compile without error?

A. `byte` B. `char`

C. `double` D. `short`

E. more than one of these

```
int x = 99;
< *1 >y = x;
```

QUESTION 34

What is the BigO notation for the `for` loop at the right ?

A. $O(n^2)$

B. $O(\log_2 n + n)$

C. $O(1)$

D. $O(n)$

E. $O(n \log_2 n)$

```
LinkedList s = new LinkedList();
//code to add n items to s not shown

Set t = new TreeSet();
for(int i=0; i<s.size(); i++) {
    t.add(s.get(i));
}
```

QUESTION 35

What is the output?

A. `[8, 5, 3]`

B. `[6, 8, 5]`

C. `[6, 8, 5, 3]`

D. `[3, 5, 8, 6]`

E. `[8, 5]`

```
Stack<Integer> s = new Stack<Integer>();
s.push(6);
s.push(8);
s.push(5);
s.push(3);
s.pop();
s.peek();
out.println(s);
```

QUESTION 36

Which of the following java expressions are equivalent to the mathematical expression: $X = \frac{2 - AB}{C^2 + D^2}$?

A. `X = 2-AB/(math.pow(C,2)+ math.pow(D,2))`
B. `X = (2-AB)/(math.pow(C,2)+ math.pow(D,2))`
C. `X = (2-A*B)/(math.pow(C,2)+ math.pow(D,2))`
D. `X = (2-A*B)/(math.sqr(C)+ math.sqr(D,2))`
E. More than one of these

QUESTION 37

In order for class `Hacker` to be syntactically correct, how many methods must class `Hacker` contain?

A. `1` B. `2` C. `3` D. `4` E. `5`

QUESTION 38

Which of the following would be legal statements?

A. `Leetable a = new Leetable();`
B. `Leetable a = new Hacker();`
C. `SuperHacker a = new Hacker();`
D. `SuperHacker a = new Leetable();`
E. more than one of these

QUESTION 39

Which of the following statements could be placed inside of class `SuperHacker`?

A. `public SuperHacker(){ }`
B. `private SuperHacker(){ }`
C. `protected SuperHacker(){ }`
D. `SuperHacker(){ }`
E. all of these

QUESTION 40

How many instance variables does interface `Leetable` contain?

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

```
public interface Leetable {
   public boolean canHack();
   public boolean didHack();
   public boolean gonnaHack();
   public static final int x=999;
}
```

```
public class Hacker implements Leetable {
    //code not shown
}
```

```
public class SuperHacker extends Hacker {
   //code not shown
}
```