Note: Correct responses are based on Java, J2sdk v 7.0, 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

What is 1001_2 minus 0001_2 ?

A. 1001_2 B. 64_{10} C. 16_{10} D. 10_8 E. 1010_2

QUESTION 2

What is output by the code to the right?

A. 0 B. 19 C. -15 D. 4 E. 23

```
int a = 19;
int b = 4;
a = b - a;
out.println(a);
```

QUESTION 3

What is output by the code to the right?

A. c B. 0 C. 0.0 D. 8.3 E. 8.4

```
double c = 4;
c += 4.3;
System.out.println(c);
```

QUESTION 4

What is output by the code to the right?

A. 4.0 B. 7.0 C. 5.0 D. 4 E. 5

```
double d = Math.min(7,5);
d = Math.min(d,4);
out.println(d);
```

QUESTION 5

What is output by the code to the right?

A. 32 B. 35 C. 3 D. 2.1 E. 40

```
int e = 8*4+3-2/4;
System.out.println(e);
```

QUESTION 6

What is output by the code to the right?

A. a+ B. practicesite C. site
D. a+_ E. practice

```
String f = "a+_practicesite_rocks";
f=f.substring(0,f.indexOf('_')+1);
System.out.println(f);
```

QUESTION 7

What is output by the code to the right?

A. -10 B. -11 C. 11 D. 12 E. -12

```
String g = "www.apluscompsci.com";
String h = "www.moodle.com";
System.out.println( g.compareTo(h) );
```

QUESTION 8

What is output by the code to the right?

A. 65 B. 32 C. 97 D. 67 E. 99

```
int i = 'a';
i = i + 2;
System.out.println(i);
```

QUESTION 9

What is output by the code to the right?

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

```
int j=9;
while(j>-2)
  j-=3;
  System.out.println(j);
```

QUESTION 10

What is output by the code to the right?

A. 0 B. 4 C. 12 D. 5 E. 8

```
int[] k = new int[5];
int[] m = k;
m[2] = 4;
m[7/2] = 8;
System.out.println(k[2] + m[3]);
```

QUESTION 11

What is the ***last*** value output by the code to the right ?

A. `7.50` B. `12.25` C. `10.50` D. `12.00` E. `13.5`

```
for(double p=0; p<12.75; p+=1.75)
   System.out.println(p);
```

QUESTION 12

What is output by the code to the right?

A. `0` B. `1` C. `false` D. `true` E. `n`

```
boolean n=false;
boolean o=true;
out.println((!(n||o)&&(n||o)));
```

QUESTION 13

What is output by the code to the right?

A. `12` B. `20` C. `17` D. `19` E. `23`

```
int q=17;
int r=23;
System.out.println(q|r^q&r);
```

QUESTION 14

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

A. `Double` B. `Byte` C. `int`

D. A and B only E. A, B, and C

```
ArrayList<  <*1>  > aList;
aList = new ArrayList<  <*1>    >();
```

QUESTION 15

What replaces **<*1>** in the code to the right so method `test` will return true if parameter `x` is at least 7 times as large as parameter y?

A. `return x*y>7;`
B. `return x>y*7;`
C. `return x>=y*7;`
D. `return y>=x/7;`
E. more than one of these

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

QUESTION 16

What is output by the code to the right?

A. `0` B. `1` C. `false` D. `true` E. `n`

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

QUESTION 17

What is output by the code to the right?

A. `9.79` B. `9.78` C. `9.785` D. `9.80` E. `9.8`

```
double u = 9.785f;
String v = String.format("%.2f",u);
out.println(v);
```

QUESTION 18

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

A. `[4.2, 5.7, 6.0]`
B. `[6.0, 5.7, 4.2]`
C. `[4.2, 6.0]`
D. `[5.7, 6.0]`
E. `[6.0, 4.2, 5.7]`

QUESTION 19

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

A. `0.0`
B. `4.2`
C. `6.0`
D. `5.7`
E. There is no output due to a syntax error.

```
Stack<Double> w;
w = new Stack<Double>();
w.add(4.2);
w.push(5.7);
w.add(6.0);
out.println(w);             // line 1
w.pop();
w.pop();
out.println(w.peek());      // line 2
```

QUESTION 20

Which of the following implements `List`?

A. `LinkedList` B. `ArrayList` C. `Stack` D. `Queue` E. more than one of these

QUESTION 21

Which of the following does ***not*** implement the `Comparable` interface?

A. `Double` B. `Byte` C. `String` D. `Float` E. `ArrayList`

QUESTION 22

What is output by the code to the right?

A. `0` B. `-1` C. `6` D. `-6` E. `1`

```
Long x;
x = new Long("090");
Long y;
y = new Long("096");
out.println(x.compareTo(y));
```

QUESTION 23

What is output by the code to the right?

A. `64` B. `128` C. `512` D. `1024` E. `1`

```
out.println( 256 & 128 << 2 ^ 64 );
```

QUESTION 24

What is the output by the code to the right?

A. `falsefalsefalse` B. `falsetruefalse`

C. `truefalsetrue` D. `truetruetrue`

E. There is no output due to a syntax error.

```
int[] x = {1,2,3};
int[] y = {1,2,3};
out.print( x instanceof Object );
out.print( x.equals(y) );
out.print( Arrays.equals(x,y) );
```

QUESTION 25

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

A. `3` B. `5` C. `9` D. `1` E. `0`

QUESTION 26

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

A. `3` B. `5` C. `9` D. `1` E. `0`

```
Queue<Integer> theQ;
theQ = new PriorityQueue<Integer>();
theQ.add(3);
theQ.add(5);
theQ.add(1);
theQ.add(9);
out.println(theQ.remove()); // line 1
theQ.remove();
theQ.add(0);
theQ.remove();
out.println(theQ.remove()); // line 2
```

QUESTION 27

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

A. `jump shot` B. `power play` C. `dunk`

D. `free throw` E. `field goal`

QUESTION 28

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

A. `touchdown` B. `power play` C. `null`

D. `jump shot` E. `field goal`

QUESTION 29

What is the output of `// line 3`?

A. `3` B. `4` C. `5` D. `6` E. `7`

```
Map<Integer, String> sports;
sports = new TreeMap<Integer, String>();
sports.put(3,"field goal");
sports.put(2,"dunk");
sports.put(2,"jump shot");
sports.put(1,"extra point");
sports.put(1,"free throw");
sports.put(2,"power play ");

out.println(sports.get(2));   // line 1

out.println(sports.get(5));   // line 2

out.println(sports.size());   // line 3
```

QUESTION 30

Assuming a binary search tree has 7 levels, what is the ***minimum*** number of leaves that it could have?

A. 5 B. 1 C. 2 D. 3 E. 7

QUESTION 31

Which of the following methods is called when attempting to find the position of a value in a String?

A. `substring()` B. `contains()` C. `indexOf()` D. `charAt()` E. more than one of these

QUESTION 32

What replaces **<*1>** in the code to the right so that method `isMystery` will run as efficiently as possible?

A. `i<Math.sqrt(num)`

B. `i<num`

C. `i<=Math.sqrt(num)`

D. A and B only

E. A,B, and C

QUESTION 33

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

A. is num an even number

B. is num an odd number

C. is num a perfect number

D. is num a prime number

E. is num a psuedo random number

QUESTION 34

What is the runtime efficiency of `isMystery`? Choose the most restrictive correct answer.

A. $O(\log_2 N)$

B. $O(N)$

C. $O(N^2)$

D. $O(N)*O(\log_2 N)$

E. $O(N^2)*O(\log_2 N)$

QUESTION 35

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

A. `111 is not something`

B. `111 is something`

C. `111 something.`

D. `111 not something.`

E. There is no output due to a syntax error.

```
public class Mystery
{
  private int num;

  public Mystery(int n)
  {
    setNum(n);
  }

  public void setNum(int n)
  {
    num=n;
  }

  public boolean isMystery()
  {
    boolean check = true;
    for(int i=2;   <*1>   ;i++){
       if(num%i == 0)
          return false;
     }
    return true;
  }

  public String toString()
  {
    if(isMystery())
       return ""+num+" is something.";
    else
       return ""+num+" is not something.";
  }
}


////////////////////////////////////////
//client code

Mystery runner = new Mystery(111);
out.println(runner);     //line 1
```

QUESTION 36

What replaces **<*1>** in the code to the right so the default `Matrix` constructor would be complete?

A. `mat = new int[][];`

B. `mat = new int[0][];`

C. `mat = new int[][0];`

D. `mat = new int[0];`

E. more than one of these

QUESTION 37

What replaces **<*2>** in the code to the right so the `Matrix` initialization constructor would correctly use the given parameters to instantiate a new matrix?

A. `mat = new int[rows][cols];`

B. `mat = new int[c][rows];`

C. `mat = new int[r][c];`

D. A and B

E. A, B, and C

QUESTION 38

What replaces **<*3>** in the code to the right so that the size of `one` and the size of the receiving row of `mat` will be the same?

A. `setRowSize(r,one.length());`

B. `setRowSize(r,one.size());`

C. `setRowSize(r,one.length);`

D. `setRowSize(one.size(),r);`

E. `setRowSize(one.length,r);`

QUESTION 39

What replaces **<*4>** in the code to the right so that the `for` loop would correctly access all rows in `mat`?

A. `int row : mat`

B. `int[] row : mat[][]`

C. `int[] row : mat[r]`

D. `int[] row : mat[r][c]`

E. `int[] row : mat`

QUESTION 40

Assume all blanks have been filled correctly. After executing `//line 1`, how many rows does instance variable `mat` contain?

A. 3 B. 6

C. 0 D. 2

E. 4

```
public class Matrix
{
   private int[][] mat;

   public Matrix() {
      <*1>
   }
   public Matrix(int rows, int cols){
      <*2>
   }

   public void setRowSize(int r, int size)
   {
     mat[r]=new int[size];
   }

   public void setRow(int r, int[] one)
   {
      <*3>
      for(int i=0; i<one.length; i++)
        mat[r][i]=one[i];
   }

   public String toString()
   {
     String output = "";
     for(  <*4>  ){
       for(int spot : row)
         output+=spot;
       output+="\n";
     }
     return output;
   }
}


//test code in the main of another class

Matrix demo = new Matrix(2,11);   //line 1
```