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

$1110_2 - 1010_2 = ?$

A. 101_2 B. 5_{10} C. 11_6 D. 4_{16} E. 111_3

QUESTION 2

What is output by the code to the right?

A. 23 B. L C. 24 D. 23L E. error

```
long whoot = 23L;
out.println(whoot);
```

QUESTION 3

What is output by the code to the right?

A. 49 B. 0 C. 1 D. 48 E. 2

```
char c = 49;
out.println(c);
```

QUESTION 4

What is output by the code to the right?

A. 16.0 B. 64.0 C. 12 D. 256.0 E. 4

```
double d = Math.pow(4,4);
out.println(d);
```

QUESTION 5

What is output by the code to the right?

A. 0 B. 6 C. 4 D. 2 E. 5

```
int e = (7*2/3+8%3);
out.println(e);
```

QUESTION 6

What is output by the code to the right?

A. -1 B. 4 C. 6 D. 10 E. 8

```
String f = "on my way to state";
out.println(f.lastIndexOf("y"));
```

QUESTION 7

What is output by the code to the right?

A. 0 B. 1 C. true D. false E. -1

```
String g = "yo";
String h = new String("yo");
out.println(g==h);
```

QUESTION 8

What is output by the code to the right?

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

```
int i = 256;
i = 512/i;
out.println(i);
```

QUESTION 9

What is returned by the call `dude(14)` ?

A. 28.0 B. 25.0
C. 16.0 D. 12.0
E. runtime error

QUESTION 10

What is returned by the call `dude(8)` ?

A. 28.0 B. 25.0
C. 16.0 D. 12.0
E. runtime error

```
public double dude(int whoa)
{
  if(whoa<1)
    return 1;
  else
    return dude(whoa - 5) + whoa;
}
```

QUESTION 11

Which of the following methods can be used to set decimal places?

A. print() B. printf() C. println() D. printformat() E. printfln()

QUESTION 12

What is output by the code to the right?

A. 4 B. 4.0 C. 2.0 D. 8.0 E. error

```
double n = 8.0f;
float o = 2.0;
out.println(n/o);
```

QUESTION 13

What is output by the code to the right?

A. 2 B. 3 C. 1

D. [1, 2, 3] E. [3, 2, 1]

```
List<Integer> p;
p = new ArrayList<Integer>();
p.add(2);
p.add(3);
p.add(0,1);
out.println(p);
```

QUESTION 14

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

A. Collection B. LinkedList C. Map

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

```
<*1> <Double> that;
that = new  <*1>  <Double>();
```

QUESTION 15

What is output by the code to the right?

A. 36 B. 14 C. 44 D. 41 E. 53

```
out.println(11 << 2 | 33 >> 1 & 8);
```

QUESTION 16

What is returned by the call `why(2,3)`?

A. 1 B. 2

C. 3 D. 4

E. more than one of these

```
public int why(int a, int b){
  if(a%b==0)  return 1;
  else if(a%b==1)  return 2;
  else if(a%b==2)  return 3;
  return 4;
}
```

QUESTION 17

What is output by the code to the right?

A. 341 B. 34.01 C. 1

D. 4.0 E. error

```
Object[] u = {3,Math.ceil(3.1),"1"};
String theSum="";
for(Object v : u)
  theSum += v;
out.println(theSum);
```

QUESTION 18

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

A. String word : chunks

B. word : chunks

C. String word : pieces

D. word : pieces

E. String word : output

```
String output="";
String it = "[to]";
String chunks = "on my way to state";
String[] pieces = chunks.split(it);
for(  <*1>  )
{
  if(word.matches(".*y.*"))
    output = output + word;
}
out.println(output);         // line 1
out.println(pieces.length);  // line 2
```

QUESTION 19

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

A. n my way B. my way C. on my

D. on my way to E. on my way state

QUESTION 20

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

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

QUESTION 21

Which of the following is ***not*** a subinterface of Collection?

A. List   B. Map   C. Set   D. SortedSet   E. Queue

QUESTION 22

What is output by the code to the right?

A. 23   B. 23*2   C. 46

D. 232   E. error

```
Object goHead = new Short("23");
goHead = (Short)goHead * 2;
out.println(goHead);
```

QUESTION 23

What is the output of // line 1?

A. 2   B. 3

C. 4   D. 5

E. 0

QUESTION 24

What is the output of // line 2?

A. 2   B. 3

C. 4   D. 5

E. 0

```
public void fun(int[][] mat){
  for(int r=0; r<mat.length; r++)
    for(int c=0; c<=r; c++)
       mat[c][r] = c * c + r * r;
}

//test code
int[][] m = {{1,2,3},{1,2,3},{1,2,3}};
fun(m);
out.println(m[1][2]+m[0][0]);   // line 1
out.println(m[2][1]+m[1][1]);   // line 2
```

QUESTION 25

What is output by the code to the right?

A. 0   B. 1   C. 2   D. 5   E. 3

```
int j = 10;
if(j%2==0)
  out.println(j/2);
else if(j%2==1)
  out.println(j/3);
```

QUESTION 26

What is output by the code to the right?

A. 32   B. 20   C. 22   D. 25   E. 28

```
int what=0;
for(int k=1; k<25; k+=2){
  if(k/3==1||k/3==3)
    what=what+k;
}
out.println(what);
```

QUESTION 27

What is output by the code to the right?

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

```
boolean first = true;
boolean last = false;
out.println(!first && !last || last);
```

QUESTION 28

Insert the following numbers into a binary search tree in the order listed. 88 19 11 34 56 32 45 22 12
How many leaves would the binary search tree contain?

A. 2   B. 3   C. 4   D. 5   E. 6

QUESTION 29

Insert the following numbers into a binary search tree in the order listed. 88 19 11 34 56 32 45 22 12
How many levels would the binary search tree contain?

A. 2   B. 3   C. 4   D. 5   E. 6

QUESTION 30

Which of the following is a quadratic algorithm?

A. merge sort B. linear search C. binary search D. selection sort E. quick sort

QUESTION 31

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

A. if all items in ascending order

B. if all items in descending order

C. if all items the same

D. if all items integers

E. if all items bigger than spot 0

QUESTION 32

What is method `doWhat` doing to `stuff`?

A. putting all odd values first

B. putting all values in ascending order

C. putting all values in descending order

D. putting all even values first

E. putting all null values first

QUESTION 33

What is the best case runtime efficiency of `doWhat`? Choose the most restrictive correct answer.

A. O(N)

B. $O(N^2)$

C. $O(\log_2 N)$

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

E. O(1)

QUESTION 34

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

A. `0` B. `1`

C. `true` D. `false`

E. syntax error

QUESTION 35

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

A. `0` B. `1`

C. `true` D. `false`

E. syntax error

```
class What
{
  public boolean isWhat(Comparable[] stuff)
  {
    for(int i=0; i<stuff.length-1; i++){
       if(stuff[i].compareTo(stuff[i+1]) < 1)
          return false;
     }
    return true;
  }

  public void doWhat(Comparable[] stuff)
  {
    if(isWhat(stuff)) return;

    for(int i=0;i<stuff.length-1;i++)
    {
      int spot=i;
      for(int j=i;j<stuff.length;j++){
        if(stuff[j].compareTo(stuff[spot])>0)
          spot=j;
      }
      if(spot==i) continue;
      Comparable save=stuff[i];
      stuff[i]=stuff[spot];
      stuff[spot]=save;
    }
  }
}
```

```
//test code in the main of another class
What why = new What();
Comparable[] list = {3,2,5,6,1,9};
out.println(why.isWhat(list));   // line 1
why.doWhat(list);
out.println(why.isWhat(list));   // line 2
```

QUESTION 36

What replaces **<*1>** in the code to the right so that `Buzzard` and `Albatross` would be descendants of `Bird`?

A. `implements` B. `inherits` C. `super`

D. `extends` E. A and B only

QUESTION 37

Assume **<*1>** was replaced correctly. What replaces **<*2>** in the code to the right so the `Flock add()` method would be complete?

A. `flyingV.put(bird);`

B. `flyingV.set(bird);`

C. `flyingV.add(bird);`

D. `flyingV.add(Bird);`

E. more than one of these

QUESTION 38

Assume **<*1>** and **<*2>** were replaced correctly. What replaces **<*3>** in the code to the right so that the `allFly()` method would access all birds in the flock?

A. `Bird b : flyingV`

B. `Buzzard b : flyingV`

C. `Albatross b : flyingV`

D. A and B

E. A, B, and C

QUESTION 39

Assume all previous blanks were replaced correctly. What replaces **<*4>** in the code to the right so that the `allFly()` method would make all birds in the flock fly?

A. `b.fly();`

B. `(Bird)b.allFly();`

C. `b.allFly()`

D. `b.fly;`

E. more than one of these

QUESTION 40

What is the worst case runtime efficiency of `allFly`? Choose the most restrictive correct answer. Assume method `fly` has a constant runtime.

A. $O(N)$

B. $O(N^2)$

C. $O(log_2N)$

D. $O(N)*O(log_2N)$

E. $O(1)$

```
abstract class Bird
{
  public abstract void fly();
}

class Buzzard <*1>  Bird
{
  public void fly(){
    //code not shown
  }

  public String toString(){
     //code not shown
  }
}

class Albatross <*1>  Bird
{
  public void fly(){
    //code not shown
  }

  public String toString(){
     //code not shown
  }
}

class Flock
{
  private LinkedList<Bird> flyingV;

  public Flock(int size){
    //code not shown
  }

  //adds another bird to the flock
  public void add(Bird bird){
    <*2>
  }

  //makes all birds in the flock fly
  public void allFly(){
     for( <*3> ){
        <*4>
     }
  }

  public String toString(){
     //code not shown
  }
}
```