

QUESTION 1

What is the sum of 512_8 and 177_8 ?

- A. 601_8 B. 701_8 C. 711_8 D. 611_8 E. 612_8

QUESTION 2

What is output by the code to the right?

- A. 10.0 B. 7.0 C. 2.5
D. 9.0 E. 8.0

```
double a = 2.5;
double b = 2.0;
a *= b + 2;
System.out.println( a );
```

QUESTION 3

What is output by the code to the right?

- A. 12 B. 20 C. 2
D. 22 E. 11

```
int sum = 0;
for(int i = 1; i < 12; i++){
    sum += 2;
}
System.out.print( sum );
```

QUESTION 4

What is output by the code to the right?

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

```
String s1 = "A";
String s2 = "B";
System.out.print( s1.compareTo( s2 ) );
```

QUESTION 5

What is output by the code to the right?

- A. 2 B. 0 C. 7
D. 1 E. -1

```
int[] scs = {3, 1, 0, 2, 3, 0, 1};
System.out.print( scs[ scs[0] ] );
```

QUESTION 6

What is output by the code to the right?

- A. 4 B. 7.5 C. 3
D. 2 E. 7

```
int r = 3;
int s = 2;
int t = r * s + r / s;
System.out.print( t );
```

QUESTION 7

What is output by the code to the right?

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

```
boolean p = true;
boolean q = !p;
System.out.print( p && !q );
System.out.print( " " );
System.out.print( q || !p );
```

<p>QUESTION 8</p> <p>What is output by the code to the right?</p> <p>A. 12 B. 2 C. 1</p> <p>D. 21 E. 212</p>	<pre>double m = 1.5; double n = 2.5; if(m > n) n *= 2; else m *= 2; if(m > 2) System.out.print(1); else System.out.print(2);</pre>
<p>QUESTION 9</p> <p>Consider the <code>Person</code> class and client code to the right . What is output by the statement marked line 1?</p> <p>A. 0_0</p> <p>B. null_null</p> <p>C. 150_70</p> <p>D. 70_150</p> <p>E. p1</p>	<pre>public class Person{ private int height; private int weight; public Person(){ this(70, 150); } public Person(int h){ height = h; } public Person(int h, int w){ height = h; weight = w; } public String toString(){ return height + "_" + weight; } }</pre>
<p>QUESTION 10</p> <p>Consider the <code>Person</code> class and client code to the right. What is output by the statement marked line 2?</p> <p>A. 0_0</p> <p>B. p2</p> <p>C. null_null</p> <p>D. 54_150</p> <p>E. 54_0</p>	<pre>//////////////////////////////////// // client code Person p1 = new Person(); System.out.println(p1); // line 1 Person p2 = new Person(54); System.out.println(p2); // line 2</pre>
<p>QUESTION 11</p> <p>What is output by the code to the right?</p> <p>A. 14 B. 58 C. -58</p> <p>D. 3364 E. 232</p>	<pre>int m = 58; int n = m >> 2; System.out.print(n);</pre>
<p>QUESTION 12</p> <p>What is output by the code to the right?</p> <p>A. 5 B. 2 C. 0</p> <p>D. 20 E. 10</p>	<pre>int x = 10; System.out.print(Math.max(x, (x / 2)));</pre>

<p>QUESTION 13</p> <p>What is output by the code to the right?</p> <p>A. AlanKay B. AlannKay</p> <p>C. AlanKAY D. Alan Kay</p> <p>E. Alan Kay</p>	<pre>String name = "Alan\nKay"; System.out.print(name);</pre>
<p>QUESTION 14</p> <p>What is output by the code to the right?</p> <p>A. 275.000 B. 275 C. +300</p> <p>D. +275 E. +000275</p>	<pre>System.out.printf("%+3d", 275);</pre>
<p>QUESTION 15</p> <p>What is returned by the method call <code>process(-2)</code>?</p> <p>A. -5 B. 3 C. -3</p> <p>D. 5 E. -2</p>	<pre>public int process(int z){ final int LOCAL = z * 2; z++; z = z + LOCAL; return z; }</pre>
<p>QUESTION 16</p> <p>What is output by the code to the right?</p> <p>A. 1 B. 0 C. 2</p> <p>D. 7 E. 4</p>	<pre>String stuff = "two three five seven"; String[] words = stuff.split("\\s+"); System.out.print(words.length);</pre>
<p>QUESTION 17</p> <p>What is output by the code to the right?</p> <p>A. 0123 B. 1234 C. 123</p> <p>D. 0000 E. 1123</p>	<pre>int[] fibs = {1, 1, 2, 3}; for(int i : fibs) System.out.print(i);</pre>
<p>QUESTION 18</p> <p>What replaces <*1> in the code to the right so that the code segment compiles without error?</p> <p>A. (String) B. (Object)</p> <p>C. (length) D. (toString)</p> <p>E. More than one of these is correct.</p>	<pre>Object obj = "Sam"; int len = (<*1> obj).length();</pre>
<p>QUESTION 19</p> <p>What is returned by the method call <code>recurs(7)</code>?</p> <p>A. 10 B. 2 C. 4</p> <p>D. 7 E. 20</p>	<pre>public int recurs(int n){ int result = 0; if(n <= 3) result = 2; else result = recurs(n - 2) + (n - 2); return result; }</pre>

<p>QUESTION 20</p> <p>What is output by the code to the right?</p> <p>A. bfb B. bbfb C. b D. bbbfb E. bfb</p>	<pre>for(int i = 8; i < 13; i++){ if(i % 3 != 0 && i % 5 != 0) continue; if(i % 5 == 0) System.out.print('f'); System.out.print('b'); }</pre>
<p>QUESTION 21</p> <p>What is output by the client code to the right?</p> <p>A. -3 B. -8 C. -6 D. 0 E. -7</p>	<pre>public int off(int month){ int result = -4; switch(month){ case 1: result = -3; break; case 3: case 5: case 8: case 10: result = -1; break; default: result = 0; } return result; } // client code System.out.print(off(1) + off(7));</pre>
<p>QUESTION 22</p> <p>What is output by the code to the right?</p> <p>A. 0 B. 10 C. null D. There is no output due to a syntax error. E. There is no output due to a runtime error.</p>	<pre>List<String> titles = new List<String>(); System.out.print(titles.size());</pre>
<p>QUESTION 23</p> <p>What is output by the code to the right?</p> <p>A. 1 B. 2 C. 12 D. There is no output due to a syntax error. E. There is no output due to a runtime error.</p>	<pre>int[] ps = {2, 3, 5, 7, 11}; if(ps[3] < ps.length && ps[ps[3]] > 0) System.out.print(2); else System.out.print(1);</pre>
<p>QUESTION 24</p> <p>Which of the following best describes the purpose of an <code>Iterator</code> object?</p> <p>A. Provide a way to insert elements into a data structure. B. Provide access to the <code>private</code> instance variables of a data structure and a way to change their capacity. C. Provide a standard way to access the elements of a data structure one element at a time. D. Provide a way for data structures to hold any type of object. E. Provide a way to sort all the elements of a data structure.</p>	

QUESTION 25

What replaces **<*1>** in the code to the right so that the body of the while loop is skipped if char c has been found in String s?

- A. result
- B. !result
- C. result == -1
- D. result != -1
- E. continue

Assume **<*1>** is filled in correctly.

QUESTION 26

Which searching algorithm does method findChar use?

- A. hash B. binary C. tree
- D. heap E. sequential

```
public int findChar(String s,
                    char c,
                    int start){
    int result = -1;
    int index = start;
    while( <*1> && index < s.length() ){
        if( s.charAt(index) == c )
            result = index;
        index++;
    }
    return result;
}
```

QUESTION 27

Which of the following is a Java keyword?

- A. do B. foreach C. try D. extra E. args

QUESTION 28

What is output by the code to the right?

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

```
int x = 3;
int y = 5;
if( (x > y) && (x == y) || (x * 2 > y) )
    System.out.print(1);
else
    System.out.print(2);
```

QUESTION 29

Consider method divide to the right. When the code is executing, if the lines marked Point A and Point B are reached, is the Boolean expression `n % 3 == 0` never, sometimes, or always true at those points?

	Point A	Point B
A.	Always	Always
B.	Always	Never
C.	Sometimes	Sometimes
D.	Sometimes	Never
E.	Always	Sometimes

```
public void divide(int n){
    if( n > 0 ){
        while( n % 3 == 0 ){
            // Point A
            n = n / 3;
            // Point B
        }
    }
    System.out.print( n );
}
```

QUESTION 30

In the code to the right how many times is the Boolean expression `i < vals.length` evaluated?

- A. `vals.length2` B. `vals.length - 1`
 C. `vals.length` D. `vals.length + 1`
 E. `vals.length / 2`

```
// pre: vals.length > 0
public int look(int[] vals, int find){
    int count = 0;
    for(int i = 0; i < vals.length; i++){
        count++;
        if( vals[i] == find )
            count--;
    }
    return count;
}
```

QUESTION 31

Assume `vals.length` is even. If exactly half of the elements in `vals.length` are equal to the value stored in the variable `find` what will the value returned by method `look` equal?

- A. `vals.length` B. `0`
 C. `(vals.length/2)` D. `1`
 E. `-(vals.length/2)`

QUESTION 32

The following values are inserted one at a time into a binary search tree using the traditional insertion algorithm. What is the result of an in-order traversal of the resulting tree?

5, 12, 0, -3, 9

- A. -3 0 5 9 12 B. 5 12 0 -3 9 C. 12 9 5 0 -3
 D. 0 -3 5 9 12 E. 5 0 -3 9 12

QUESTION 33

Given the following measurements, what is the most likely running time for method `sample(int[] data)` where `N` is equal to `data.length`? Choose the most restrictive correct answer.

Value of N Time for method `sample` to complete

2,000 1 second
 4,000 2 seconds
 6,000 3 seconds

- A. $O(N)$ B. $O(N \log N)$ C. $O(N^2)$ D. $O(1)$ E. $O(N^{3/2})$

QUESTION 34

What replaces `<*1>` in the code to the right to place the value stored in the variable `x` at the end of `data` if the Boolean expression `x % 2 == 0` is true?

- I. `data.add(x)`
 II. `data.addLast(x)`
 III. `x = data.removeFirst()`
 A. I only B. II only C. III only
 D. I and II E. I, II, and III

```
public void test(LinkedList<Integer> data,
                int x){
    if(x % 2 == 0){
        <*1>;
    }
}
```

QUESTION 35

Which sorting algorithm do the two methods to the right named `sort` implement?

- A. merge sort
- B. selection sort
- C. bubble sort
- D. quicksort
- E. insertion sort

```
public void sort(int[] data){
    int[] temp = new int[data.length];
    sort(data, temp, 0, data.length - 1);
}

public void sort(int[] data,
                int[] temp, int i, int j){
    if(i < j){
        int mid = (i + j) / 2;
        sort(data, temp, i, mid);
        sort(data, temp, mid + 1, j);

        int le = mid;
        int tp = i;
        int ne = j - i + 1;
        while( (i <= le) && (mid + 1 <= j) ){
            if( data[i] <= data[mid + 1] )
                temp[tp] = data[i++];
            else
                temp[tp] = data[mid++ + 1];
            tp++;
        }

        while( i <= le)
            temp[tp++] = data[i++];

        while( mid + 1 <= j)
            temp[tp++] = data[mid++ + 1];

        for(int k = 0; k < ne; k++){
            data[j] = temp[j];
            j--;
        }
    }
}
```

QUESTION 36

What is the Big O of the method named `sort` with a single parameter given an array of `ints` that is already sorted into ascending order? `N = data.length`. Choose the most restrictive correct answer.

- A. $O(N)$
- B. $O(N \log N)$
- C. $O(N^{3/2})$
- D. $O(N^2)$
- E. $O(N^3)$

QUESTION 37

What is output by the code to the right?

- A. 02468 B. 0 C. 10
- D. 0246810 E. 024

```
Queue<Integer> q;
q = new LinkedList<Integer>();

for(int i = 0; i < 10; i += 2)
    q.add(i);

for(int i = 0; i < q.size(); i++)
    System.out.print( q.remove() );
```

QUESTION 38

What is output by the client code to the right?

- A. frums
- B. fmrsu
- C. usrmf
- D. ffffff
- E. smurf

```
public class Structure<E>{

    LinkedList<E> con;

    public Structure(){
        con = new LinkedList<E>();
    }

    public void add(E obj){
        con.addFirst(obj);
    }

    public E access(){
        return con.getFirst();
    }

    public E remove(){
        return con.removeFirst();
    }

    public boolean isEmpty(){
        return con.size() == 0;
    }
}

// client code
Structure<Character> st;
st = new Structure<Character>();
String cartoon = "smurf";

for(int i = 0; i < cartoon.length(); i++)
    st.add( cartoon.charAt(i) );

while( !st.isEmpty() )
    System.out.print( st.remove() );
```

QUESTION 39

What type of data structure does the Structure class implement?

- A. A binary search tree
- B. A stack
- C. A priority queue
- D. A queue
- E. A linked list

QUESTION 40

What is output when method kick is called if mat is the 2D array below?

1	4	8	-5	8
3	3	8	1	0
2	0	7	7	5
-4	4	3	3	3
0	2	0	4	1

- A. 11000 B. 11111 C. 00000
- D. 00111 E. 00101

```
public void kick(int[][] mat){
    for(int i = 0; i < mat.length; i++)
        System.out.print( off(mat, i) );
}

public int off(int[][] mat, int i){
    int r = 0;
    int c = 0;
    for(int j = 0; j < mat.length; j++){
        r += mat[i][j];
        c += mat[j][i];
    }

    return (r > c) ? 0 : 1;
}
```