QUESTION 1

What is the value of $CAB_{16} + 999_{16}$?

A. $1CAA_{16}$ B. $CAB999_{16}$ C. 1644_{16} D. LJK_{16} E. None of these

QUESTION 2

Which of the following replaces **<*1>** in the code to the right to declare a constructor for the class that will build a `Test` object with all of its data members initialized to `0`?

A. `public Test():0,0,0,0 {};`

B. `public Test(x=0,y=0,c=0,d=0);`

C. `public Test();`

D. `public Test() {}`

E. None of these

```
public class Test {

  <*1>

  public double add() { return c + d; }

  public int multiply() { return x * y; }

  public void setX(int x) { this.x = x; }
  public void setY(int y) { this.y = y; }
  public void setC(double c)
                          { this.c = c; }
  public void setD(double d)
                          { this.d = d; }

  private int x, y;
  private double c, d;
}
```

For the remaining questions, assume that **<*1>** has been filled in correctly.

QUESTION 3

Which of the following creates a `Test` object and sets its `x` and `y` data members to `2`?

A.
```
Test t = new Test();
t.setX(2);
t.setY(2);
```

B.
```
Test t();
t.setX(2);
t.setY(2);
```

C.
```
Test t;
t.setX(2);
t.setY(2);
```

D.
```
Test t;
t.setX.setY(2);
```

E. More than one of these

QUESTION 4

What can access private data member `c`?

A. Any class

B. Any code in the class `Test` or in the same package

C. Any `main()` method

D. Any class that imports `Test`

E. None of these

QUESTION 5

What is the value of `count` after executing the nested loop to the right if `m` is initialized to `234` and `n` is initialized to `24`?

A. `48` B. `96`

C. `72` D. `100`

E. None of these

```
int m,n;

// code not shown to initialize m and n

int count = 0;
for (int i=0; i<m; i+=10)
  for (int j=1; j<n; j*=3)
    ++count;
```

QUESTION 6

What is the running time of the nested loop to the right? Choose the smallest correct answer.

A. `O(m log n)` B. `O(mn)`

C. `O(log m log n)` D. `O(m + n)`

E. None of these

QUESTION 7

Which of the following outputs the letter `D` and nothing else when `int i` is `65`?

A.
```
if (i>90)
 System.out.print('A');
else if (i>80)
 System.out.print('B');
else if (i>70)
 System.out.print('C');
else if (i>60)
 System.out.print('D');
else
 System.out.print('F');
```

B.
```
if (i>90)
 System.out.print('A');
if (i>80 && i<=90)
 System.out.print('B');
if (i>70 && i<=80)
 System.out.print('C');
if (i>60 && i<=70)
 System.out.print('D');
if (i<=60)
 System.out.print('F');
```

C.
```
if (i>90)
 System.out.print('A');
if (i>80)
 System.out.print('B');
if (i>70)
 System.out.print('C');
if (i>60)
 System.out.print('D');
else
 System.out.print('D');
```

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

QUESTION 8

What is the value of `array.length`?

A. `0` B. `3`

C. `4` D. `5`

E. None of these

```
int[][] array = {{1,2,3,4}, {5,6,7},
                 {8,9}, {10}};
```

QUESTION 9

What is the value of `array[1].length`?

A. `0` B. `3`

C. `4` D. `5`

E. None of these

QUESTION 10

What replaces **<*1>** and **<*2>** in the code to the right to indicate that the classes cannot be instantiated and that the methods are not being defined?

A. **<*1>:** `abstract`
<*2>: `abstract`

B. **<*1>:** `abstract`
<*2>: `virtual`

C. **<*1>:** `virtual`
<*2>: `abstract`

D. **<*1>:** `virtual`
<*2>: `virtual`

E. None of these

QUESTION 11

The volume of a pyramid is one third of the product of its height and the area of its base. What replaces **<*3>** in the code to the right to correctly compute the `volume()` method?

A. `height / 3 * base.area()`

B. `height * base.area() / 3`

C. `height / 3 * Shape2D.area()`

D. `height * Shape2D.area() / 3`

E. More than one of these

QUESTION 12

Assume that the class `Triangle` is a subclass of `Shape2D` and has a constructor which takes three integer values representing the lengths of the sides. Which of the following builds a `Pyramid p` with height `7` and an equilateral triangle base with side length `4`?

A.
```
Pyramid p(Triangle(4,4,4),7);
```

B.
```
Pyramid p = Pyramid(Triangle(4), 7);
```

C.
```
Pyramid p = new Pyramid(
            new Triangle(4), 7);
```

D.
```
Pyramid p = new Pyramid(
            new Triangle(4,4,4), 7);
```

E. None of these

```
public <*1> class Shape2D {
  public <*2> double perimeter();
  public <*2> double area();
}

public <*1> class Shape3D {
  public <*2> double volume();
  public <*2> double surfaceArea();
}

public class Pyramid extends Shape3D {
  public Pyramid(Shape2D base, int height){
    this.base = base;
    this.height = height;
  }

  public double volume() {
    return <*3>;
  }

  public double surfaceArea() {
    // code not shown
  }

  private Shape2D base;
  private int height;
}
```

QUESTION 13

What is the value of `roster[2]` after the declaration below?

```
Employee[] roster = new Employee[100];
```

A. `null`
B. undefined
C. `0`
D. invalid declaration
E. None of these

QUESTION 14

Assume part of `roster` is initialized. Which of these gives raises to all employees, ignoring empty slots?

A.
```
for (int i=0; i<100; ++i)
  roster[i].raise();
```

B.
```
roster.raise();
```

C.
```
for (int i=0; i<100; ++i)
  if (roster[i].firstName)
    roster[i].raise();
```

D.
```
for (int i=0; i<100; ++i)
  if (roster[i])
    roster[i].raise();
```

E. None of these

QUESTION 15

What is output by the code below?

```
Employee e = new Employee("Worker",
                  "Bee", 999999999,
                  5, 100000);
e.raise();
System.out.print(e.getSalary());
```

A. `5000.0`
B. `100000.0`
C. `200000.0`
D. `500000.0`
E. None of these

```
public class Employee {

  public Employee(String f, String l,
                  long id, int b, long s) {
    firstName = f;
    lastName = l;
    IDNumber = id;
    band = b;
    salary = s;
  }

  public void raise() {
    salary += salary*band/100;
  }

  public double getSalary() {
    return salary;
  }

  // other methods not shown

  private String firstName, lastName;
  private long IDNumber;
  private int band;
  private long salary;
}
```

QUESTION 16

What is returned by the following?

```
process(new StringBuffer("abcdedcba"));
```

A. 1 B. 2

C. 3 D. 4

E. None of these

```
public static int process(StringBuffer sb){
  int count = 0;
  for (int i=0; i<sb.length(); ++i)
    if (sb.charAt(i) > 'a' + count) {
      sb.setCharAt(i,
                   (char)(sb.charAt(i)+1));
      ++count;
    }
  return count;
}
```

QUESTION 17

What is output by the following?

```
StringBuffer sb =
        new StringBuffer("edcbabcde");
process(sb);
System.out.print(sb);
```

A. edcbabcde B. fedcbcdef

C. fedbabcde D. fecbabcdf

E. None of these

QUESTION 18

What is output by the code to the right on the input below?

```
hello 1
```

A. h B. hello

C. e D. hello1

E. None of these

```
// Assume getString() and getInt() are
// static methods in a class named IO that
// read a String and an integer from
// the keyboard

String s = IO.getString();
int i = IO.getInt();

try {
  System.out.print(s.charAt(i));
}
catch(StringIndexOutOfBoundsException e) {
  System.out.print("Invalid: " + i);
}
```

QUESTION 19

What is output by the code to the right on the input below?

```
hello 15
```

A. h B. Invalid: 15

C. e D. nothing

E. None of these

QUESTION 20

What does `int[] intArray` look like after the static method call `process(intArray)` when `intArray` is the array below?

1	2	3	4	5	6

A.

1	2	3	4	5	6

B.

6	5	4	3	2	1

C.

1	2	3	3	2	1

D.

6	5	4	4	5	6

E. None of these

```
public static void process(int[] A) {
  for (int i=0; i<A.length/2; ++i)
    A[A.length - i - 1] = A[i];
}
```

QUESTION 21

Which of the following replaces **<*1>** to check whether the object in position `i` of `strArray` is a `String` representing `"dog"`?

A. `(0 <= i < strArray.length) && strArray[i] == "dog"`

B. `(i>=0) && (i<strArray.length) && strArray[i] == "dog"`

C. `(0 <= i < strArray.length) && strArray[i].equals("dog")`

D. `(i>=0) && (i<strArray.length) && strArray[i].equals("dog")`

E. None of these

```
String[] strArray = new String[100];

// code to initialize all of the strings
// in the array

int i;

// code to initialize i

if (<*1>)
  // more code here
```

QUESTION 22

What is output by the code to the right?

A. `abcdefghijklmonpqrpqr`

B. `abcdefghijklmnopqr`

C. `abcdefghijklmno`

D. nothing

E. None of these

```
String[] strArray = {"abc", "def", "ghi",
                     "jkl", "mno", "pqr"};

List l = new ArrayList();

for (int i=0; i<strArray.length; ++i)
  l.add(strArray[i]);

Iterator iter = l.iterator();

while (iter.hasNext()) {
  System.out.print(iter.next());
}
```

QUESTION 23

If `nameArray` is an array of strings defined in the `main()` method of another class, which of these would be a correct call to the `MergeSort()` method from that class?

A. `MereSort(nameArray)`

B. `Sort.MergeSort(nameArray)`

C.
```
MergeSort(nameArray, 0,
                  nameArray.length)
```

D.
```
Sort.MergeSort(nameArray, 0,
                  nameArray.length)
```

E. More than one of these

QUESTION 24

Suppose the array of `Integer` objects below is sorted with the `MergeSort()` method. Which of these shows the state of the array after the two recursive calls to `MergeSort()` complete, but before the final call to `Merge()`?

17	22	8	-3	3	7

A.

17	22	8	-3	3	7

B.

8	-3	3	7	17	22

C.

-3	3	7	8	17	22

D.

8	17	22	-3	3	7

E. None of these

QUESTION 25

Which of these has the same worst case asymptotic running time as the merge sort algorithm?

A. Sequential search

B. Selection sort

C. Insertion sort

D. Quick sort

E. None of these

```
public class Sort {

  public static void MergeSort
                (Comparable[] A) {
    MergeSort(A, 0, A.length);
  }

  private static void MergeSort
      (Comparable[] A, int front,
                        int back) {
    int mid=(front+back)/2;
    if (mid==front) return;
    MergeSort(A,front,mid);
    MergeSort(A,mid,back);
    Merge(A,front,back);
  }

  private static void Merge(Comparable[] A,
                    int front, int back) {
    Comparable[] temp =
           new Comparable[back-front];
    int i=front, j=(front+back)/2, k=0;
    int mid=j;

    while(i<mid && j<back) {
      if (A[i].compareTo(A[j])<0)
        temp[k++]=A[i++];
      else
        temp[k++]=A[j++];
    }

    while (i<mid) temp[k++]=A[i++];
    while (j<back) temp[k++]=A[j++];

    for(i=0; i<back-front; ++i)
      A[front+i]=temp[i];

  }

}
```

QUESTION 26

What is output by the static method call `output(0)`?

A. nothing

B. `@`

C. `#`

D. `$`

E. None of these

QUESTION 27

What is output by the static method call `output(-300)`?

A. `!#@$@#@`

B. `!@#@$@#`

C. `#@$@#@`

D. `@#@$@#`

E. None of these

```
public static void output(int n) {
  if (n<0) {
    System.out.print('!');
    n = -n;
  }
  do {
    switch(n%3) {
    case 0: System.out.print('@');
            break;
    case 1: System.out.print('#');
            break;
    case 2: System.out.print('$');
    }
    n = n/3;
  } while (n!=0);
}
```

QUESTION 28

What is output by the static method call `output("abc123$%^456DEF")`?

A. `abc123$%^456DEF`

B. `abc123`

C. `abc123456DEF`

D. `ABC123456def`

E. None of these

```
public static void output(String s) {
  for (int i=0; i<s.length(); ++i)
    if (Character.isLetterOrDigit(
                              s.charAt(i)))
      System.out.print(s.charAt(i));
}
```

QUESTION 29

In order to do an efficient binary search of a collection of elements, which of the following properties are required?

A. The elements must be sorted

B. There must be constant time access to an element, given its index

C. The elements must be numbers

D. Both A and B

E. All of these

```
// no code for this problem
```

QUESTION 30

What is returned by the static method call `recurse(10,25)`?

A. `35`

B. `40`

C. `45`

D. `50`

E. None of these

```
public static int recurse(int i, int j) {
  if (i==0) return j;
  else return recurse(i-1, j+2);
}
```

QUESTION 31

What is output by the code below?

```
SlowQueue q = new SlowQueue(10);
q.enqueue("cat");
q.enqueue("dog");
q.enqueue("pig");
System.out.print(q.dequeue());
```

A. `dog`

B. `cat`

C. `pig`

D. `pigdogcat`

E. None of these

QUESTION 32

What is the running time of the `dequeue()` method for a `SlowQueue` object containing `n` elements? Choose the smallest correct answer.

A. `O(1)`

B. `O(log n)`

C. `O(n)`

D. $O(n^4)$

E. None of these

```
public class SlowQueue {

  public BadQueue(int maxSize) {
    size = 0;
    elements = new Object[maxSize];
  }

  public void enqueue(Object o) {
    elements[size++] = o;
  }

  public Object dequeue() {
    Object o = elements[0];
    for (int i=0; i<size; ++i)
      elements[i]=elements[i+1];
    --size;
    return o;
  }

  private int size;
  private Object[] elements;
}
```

QUESTION 33

Which of the following replaces **<*1>** to increase `sum` by the value of `A[i]`?

A. `sum ++ A[i];`

B. `sum++;`

C. `A[i] ++ sum;`

D. `sum += A[i];`

E. None of these

```
public static double average(double[] A) {
  int sum=0;
  for (int i=0; i<A.length; ++i)
    <*1>
  return sum / A.length;
}
```

QUESTION 34

Given the declarations below, which of the following expressions is true?

```
A a = new A();
B b = new B();
C c = new C();
```

A. `a instanceof B` B. `b instanceof C`

C. `a instanceof C` D. `c instanceof A`

E. More than one of these

```
public class A {
  // methods and data not shown
}

public class B extends A {
  // methods and data not shown
}

public class C extends B {
  // methods and data not shown
}
```

QUESTION 35

Suppose that the static method `f()` of class `D` takes a parameter of type `C`. Given the declarations below, which of these is a valid call to method `f()`?

```
A a = new C();
B b = new C();
C c = new C();
```

A. `D.f(a)` B. `D.f(b)`

C. `D.f(c)` D. All of these

E. None of these

QUESTION 36

What is output by the first print statement in the code to the right?

A. `true` B. `1`

C. `false` D. `0`

E. None of these

```
Character c = new Character('7');

Integer i = new Integer(7);

System.out.print(c.equals(i));

System.out.print(c.toString().
                    equals(i.toString()));
```

QUESTION 37

What is output by the second print statement in the code to the right?

A. `true` B. `1`

C. `false` D. `0`

E. None of these

QUESTION 38

Suppose the code which inputs values is not inside a `try/catch` block, and that the methods used may throw an `IOException`. Which of these could replace **<*1>** in the declaration of the `main()` method?

A. `throws IOException`

B. `throw IOException`

C. `throws new IOException`

D. `throw new IOException`

E. None of these

```
public static void main(String[] args) <*1>
{
  // input some values
  // output some results
}
```

QUESTION 39

Which of the following flips a coin named `myCoin`, and evaluates to `true` when the flip is a head?

A. `flip == HEAD`

B. `myCoin.flip == HEAD`

C. `myCoin.flip() == Coin.HEAD`

D. `myCoin.flip().HEAD`

E. None of these

QUESTION 40

Assuming that the `Random` class generates all of the integers with equal probability, what is the probability of the `flip()` method returning a head?

A. slightly over `1/2`

B. `0`

C. slightly under `1/2`

D. `1/2`

E. None of these

```
public class Coin {

  public Coin() {
    r = new java.util.Random();
  }

  public int flip() {
    int i = r.nextInt();
    if (i>=0) return HEAD;
    else return TAIL;
  }

  public static final int HEAD = 0;
  public static final int TAIL = 1;

  private java.util.Random r;
}
```