READ THIS NOW!

Failure to read and follow the instructions below may result in severe penalties.
Failure to adhere to these directions will not constitute an excuse or defense.

- Print your name in the space provided below.
- Print your name and ID number on the Opscan form; be sure to code your ID number on the Opscan form. Code Form A on the Opscan.
- Choose the single best answer for each question — some answers may be partially correct. If you mark more than one answer, it will be counted wrong.
- Unless a question involves determining whether given Java code is syntactically correct, assume that it is. The given code has been compiled and tested, except where there are deliberate errors. Unless a question specifically deals with Java import directives, you should assume the necessary library files have been imported.
- Note that questions about printed values disregard formatting completely.
- In questions/answers which require a distinction between integer and real values, integers will be represented without a decimal point, whereas real values will have a decimal point, [1054 (integer), 1054.0 (real)].
- When you have completed the test, sign the pledge at the bottom of this page and turn in the test and your Opscan.
- This is a closed-book, closed-notes examination. No calculators or other electronic devices may be used during this examination. You may not discuss (in any form: written, verbal or electronic) the content of this examination with any student who has not taken it. You must return this test form when you complete the examination. Failure to adhere to any of these restrictions is an Honor Code violation.
- There are 25 multiple-choice questions.
- Mark your answers on the test form, for future reference, and on the Opscan. The answers you mark on the Opscan form will be considered your official answers.

Do not start the test until instructed to do so!

Name ___________________________ (print: Last name, First)
VT PID: ___________________________ (print: campus email address)

Pledge: On my honor, I have neither given nor received unauthorized aid on this examination.

_____________________________  
signature
Given the following Java class declaration:

```java
public class Test2 {
    public Test2() {
        x = y = 0;
    }
    public Test2(int a, int b) {
        x = a;
        y = b;
    }
    public boolean equals(Test2 t2) {
        return (x == t2.x) && (y == t2.y);
    }
    private int x, y;
}
```

1. What, (if anything), is wrong with the declaration in `Test2` client code below?

```java
Test2 t2 = new Test2(1, 2);
```

1) The default constructor must be invoked first.
2) The second (parameterized) constructor requires `int` variable parameters.
3) The constructor must be invoked with the `new` object, (e.g., `Test2 t2 = new Test2(1, 2);`)
4) None of the above

Assuming the `Test2` class from the previous question, given the following declarations in `Test2` client code:

```java
Test2 t22 = new Test2(2, 2);
Test2 t23 = new Test2(2, 3);
```

Each of the objects contains two private int instance variables.

2. What is the total number of instance variables in the `Test2` objects `t22` and `t23`?

1) 1  2) 2   3) 3   4) 4   5) 5   6) 6   7) none of these

3. Assuming the `Test2` class above and the declarations from the previous question, what, (if anything), is wrong with the code segment in `Test2` client code below?

```java
boolean t3same = (t22.x == t23.x) && (t22.y == t23.y);
```

Why will the above code to set the `boolean t3same` variable **not** compile?

1) The `t22` instance variables must not be prefixed (or qualified) by the `t22` reference variable.
2) **The `t22` instance variables can only be accessed in the `Test2` class member functions.**
3) The “`this`” Java reserved reference variable must be used to access the instance variables.
4) None of the above

Private instance variables are encapsulated by their class.
Assuming the Test2 class from the previous questions, given the following segment in Test2 client code:

```java
Test2 t41 = new Test2( 4, 0 );
Test2 t42 = new Test2( 4, 0 );
boolean b4 = t42.equals(t41);
```

The equals member function compares the corresponding instance variable values, which are the same.

4. What is the value assigned to the boolean b4 variable?

1) true  2) false  3) none of these  4) 4

Assuming the Test2 class from the previous questions, given the following segment in Test2 client code:

```java
Test2 t51 = new Test2( 5, 0 );
Test2 t52 = new Test2( 5, 0 );
boolean b5 = (t51 == t52);
```

The equality operator compares the reference variables to determine if they are referring to the same object in memory.

5. What is the value assigned to the boolean b5 variable?

1) true  2) false  3) none of these  4) 50  5) 5

For the following 3 questions, select the value of the given Java arithmetic expression. Note that the presence of a decimal indicates a double value, rather than an int.

6) \(9 / 4 + 3.0\)  5.25  5.0

7) \(8 / 5 - 4 / 5\)  1  0.8

8) \(16 \% 4 / 3 - 2\)  -1  2  -0.666  not allowed  none of these

For the next 2 questions, select the value assigned to the relevant variable, given the declarations:

```java
int IntId;
double DecId;
```

9) \(\text{DecId} = 19 / 5;\)  3  3.0  3.8  4  none of these

10) \(\text{IntId} = (\text{int}) (5 + 2.7);\)  7  7.7  8  8.0  none of these

9) \(19 / 5\) (int division) = 3 which is converted to a double for assignment, resulting in 3.0

10) The 5 is converted to 5.0 for addition to the double 2.7 resulting in 7.7 which is truncated when typecast to an int.
11) Given the following incomplete Java code segment:

```java
String str = "3.1415";
double dval = ________________;
```

Choose the Java code below to replace the underlined blank above to convert the string into an equivalent double representation?

1) Double.doubleValue.valueOf(str)
2) Double.valueOf(str).doubleValue()
3) double.DoubleValue.valueOf(str)
4) double.valueOf(doubleValue(str))
5) None of the above

12) What is the value of the variable `Z` after the following code is executed?

```java
int W = 7, X = 3, Y = 9, Z = 0;
if (Z - X <= Y / 7) {
    Z--;  // which is true
    if (W*2-X != Y+X-1)
        Z++;  // NOT executed
    else
        Z--;
} else {
    Z = 3;
}
```

Performing the computations & substituting yields:

```java
if (-3 <= 1)  { //which is true
    Z--;  // Z = -1
    if (11 != 11) //which is false
        Z++;  // NOT executed
    else
        Z--;
} //nothing else is executed
```

1) -1  3) 0  5) 3  6) the code contains a syntax error
2) -2  4) 1  7) none of the above

13) What output will the following program produce? (Be careful this may be tricky.)

```java
int Score = 87, Rank = 2;
if (Rank > 2)
if (Score < 90)
    System.out.println("Not bad!");  //the else is matched with the previous unmatched if,
else
    System.out.println("Pretty Good!");
```

The else is matched with the previous unmatched if, which is the second if, thus the second else is entirely under control of the first if whose condition is false.

1) Not bad!  4) "Pretty Good!"
2) Pretty Good!  5) both 1 and 2
3) "Not bad!"  6) both 3 and 4
7) No output is produced.
8) none of these
14) What is the value printed for the variable Beta if the following code is executed?

```java
int Beta = 0, X = 5;
if (X % 3 == 1)
    Beta = Beta + X;
X--; // The equality operator compares the reference variables to determine if they are referring to the same object in memory.
if (X % 2 == 0)
    Beta = Beta + X;
X--; // The equality operator compares the reference variables to determine if they are referring to the same object in memory.
if (X / 3 == 0)
    Beta = Beta + X;
```

```
System.out.println("Beta = " + Beta);
```

1) 0  5) 4  8) None of these
2) 1  6) 5
3) 2  7) 6
4) 3

15) Assume that the BufferedReader reference variable myInput has been successfully associated with a FileInputStream. In the code below that attempts to read the next line of input from myInput, what will be the value assigned to the String reference variable if the `readLine` method fails.

```
String NextLine = myInput.readLine();
```

1) -1   2) 0   3) null
4) nill  5) false  6) none of the above

16) What is the value printed for the variable alpha if the following code is executed?? (Be careful here.)

```java
int num=4;
int alpha = 10;
switch (num)
{
    case 3  : alpha++;
    case 4  : alpha = alpha + 2;
    case 8  : alpha = alpha + 3;
    default  : alpha = alpha + 4;
}
System.out.println("alpha = " + alpha);
```

1) 10  2) 14  3) 12  4) 19  5) 15
Given the following Java code segment:

```java
int Count = 0;
int number = 2;
boolean done = false;
while (!done)
{
    Count++;
    number = number * 2;
    done = (number > 64);
}
System.out.println( "Count = " + Count );
```

17) What is printed out when the above code is executed?

1) 1  2) 2  3) 3  4) 4  5) 5  6) 6  7) 0  8) 10  9) none of the above

Given the following Java code segment:

```java
int i=1, sum=0, j;
while (i <= 5)
{
    sum = 0;
    j = 1;
    while (j<=i)
    {
        sum = sum + j;
        j++;
    }
    i++;
}
System.out.println( "sum = " + sum );
```

18) What is printed out for sum when the above code is executed?

1) 0  2) 1  3) 5  4) 14  5) 15  6) none of the above
19) What is the output of the following code fragment?

```java
int beta = 5;
while (beta > 1)
{
    switch (beta)
    {
        case 1 : System.out.print( 'R' );
                break;
        case 2 :
        case 4 : System.out.print( 'O' );
                break;
        case 5 : System.out.print( 'L' );
    }
    beta--;
}
System.out.println( 'X' );
```

1) X 2) ROOLX 3) LOOX 4) LOORX 5) ROOX

Given the following Java code segment:

```java
int num = 345;
while (num > 0)
{
    System.out.print(' ');
    System.out.print( num % 10 );
    num = num /10;
}
System.out.println();
```

20) What does the above code segment display?

1) 5 .5 0 2) 3 4 5
3) .5 0 0 4) 5 4 3
5) 34.5 3.45 .345 6) none of the above
For the next 5 questions, consider the incomplete static member function definition given below, assume it is located in the main program class:

```java
public static int CountAs(String aString)  //Assumes aString is not null
{
    int Counter = _________;    //Line A
    int TotalA  = 0;
    while(Counter < ________)     //Line B
    {
        if ( _________ .equals("A") )   //Line C
            TotalA=TotalA+________;   //Line D
        Counter++;
    }
    return ( _________ );     //Line E
}
```

21) How should the blank, (i.e. underscores), for the initial value of `Counter` in line A be filled?

1) -1  
2) 0  
3) 1  
4) 2  
5) it should be left blank  
6) none of these

Counter should be initialized to zero since string indicies start at 0.

22) How should the blank in line B be filled, (for the less than comparison to `Counter`)?

1) `aString.length()`  
2) `length(aString)`  
3) `aString.hasMoreElements()`  
4) `Counter.length()`  
5) none of these

The number of characters to be compared is given by the `length()`.

23) How should the blank in line C be filled?

1) `aString.substring(Counter)`  
2) `aString.substring(Counter).toUpperCase()`  
3) `aString.substring(Counter, Counter + 1)`  
4) `aString.substring(Counter, Counter + 1).toUpperCase()`  
5) none of these

The substring method allows access to each individual string character. The better answer is 4) which will also count the lowercase “a”’s.

24) How should the blank in line D be filled?

1) 1  
2) 2  
3) `Counter`  
4) `++`  
5) none of these

For every “A” character found the total should be increased by 1.

25) How should the blank in line E be filled?

1) `TotalA`  
2) `Counter`  
3) `aString.length()`  
4) none of these

TotalA is used to store the number of A characters found in the string.