

CS 107 - Introduction to Programming
Midterm Exam #1 - Prof. Reed
Spring 2009

What is your name?: _____

There are two sections:

I. True/False. 72 points; (36 questions, 2 points each)

II. Multiple Choice 28 points; (7 questions, 4 points each)

100 points total

This test is worth **10%** of your final grade. Please fill in your answers on the bubble form. After the test you may keep these pages, but you must turn in your bubble form. This test is open book and open notes. You have 50 minutes.

- For the multiple choice problems, select the *best* answer for each one and select the appropriate letter on your answer sheet.
- *Be careful* - more than one answer may seem to be correct. Many questions are tricky.
- Some problems ask you to determine whether something is *valid*. Something is *valid* if it would not generate a compiler error and would execute without the program crashing.

I. True/False: (2 points each)

- T F** 1. Java programs *must* have BlueJ to work. You can't run a Java program without BlueJ.
- T F** 2. A java class name should be the same as the filename in which it is found. For instance the *Car* class should be stored in *Car.java*
- T F** 3. A single line comment using *//* can be safely nested inside a block style comment that uses */*...*/*
- T F** 4. To change your password on the UNIX system, you need to use the *password* command.
- T F** 5. Multiple variables can be declared at one time, such as:
`int x,y,z;`
- T F** 6. Multiple variables can all be initialized on the same line, such as:
`x=y=z=0;`
- T F** 7. The output of the statements below is: 2.5
`int value = 5;
System.out.println(value / 2);`
- T F** 8. The output of the statements below is: 2.5
`int value = 5;
System.out.println((float)value / 2);`
- T F** 9. The output of the statements below is: 2.5
`int value = 5;
System.out.println(1.0 * value / 2);`

T F 10. The output of the statements below is: 2.5

```
int value = 5;
System.out.println( value / 2 * 1.0);
```

T F 11. The output of the statements below is: 2.5

```
double value = 5.0;
System.out.println( value / 2);
```

T F 12. The following statements are *valid* (compile and run) in Java:

```
int x = 0;
x = x + 1;
```

T F 13. The following statements are *valid* (compile and run) in Java:

```
int x = 0;
x + 1 = x;
```

T F 14. It is possible in Java to write one line of code that generates 3 lines of output.

T F 15. It is possible in Java to write 3 lines of code that generates 1 line of output.

T F 16. A convenient way to line up different magnitude numbers in a table involves using *printf*.

T F 17. Multi-word variable names in Java should have the first letter of each word capitalized, except for the first word.

T F 18. By convention, global constants *must* be in all capital letters in Java, otherwise the compiler will generate an error.

T F 19. When creating an instance of the Scanner class, it must be called *keyboard*.

For the following 3 statements, tell whether or not each statement is *valid* (compiles & runs).

T F 20. `int whiLe = true;`

T F 21. `char initials = "A";`

T F 22. `int pie = (int)6.28/2;`

T F 23. Consider writing multiple if-else-if statements to handle assigning a letter grade (e.g. 'A', or 'B') given a numerical grade (e.g. 93, or 82). The if-else-if statements should be indented inside the if statement above.

T F 24. Consider writing multiple if-else-if statements to handle menu options. The if-else-if statements should be indented inside the if statement above.

T F 25. The following statements compile and run in Java:

```
String name = "Englebert";
if( name == "Eeagle Bert") {
    System.out.println("Are equal");
}
```

T F 26. To check and see if the value of variable x is between 3 and 9 we could use:

```
int x = 4;
if( 3 < x < 9) {
    System.out.println("x is between 3 and 9");
}
```

T F 27. The output of the statement below is: -2

```
System.out.println( 8 % -3);
```

T F 28. After running the code shown below, the value stored in variable x is: 30

```
int y = 5;
int x = y * y++;
```

T F 29. The output of the following lines of code is: Done End

```
boolean notDone = false;
if (notDone = true)
    System.out.println("not Done ");
if( notDone = false)
    System.out.println("Done ");
else
    System.out.println("Undecided ");
System.out.println("End");
```

T F 30. The output of the program segment below is: 3

```
String jumbo = "1";
String shrimp = "2";
System.out.println(jumbo + shrimp);
```

T F 31. The output of the following statements is: 25 Done

```
int x = 2;
int y = 5;
System.out.print( " " + x + y);
System.out.println(" Done");
```

T F 32. The output of the following statements is: 25 Done

```
int x = 2;
int y = 5;
System.out.print( x + y + " ");
System.out.println(" Done");
```

T F 33. The following code is valid (compiles and runs) in Java:

```
int x,y;
for( x=1, y=7; x<y; x++,y--)
    System.out.println(x*y);
```

T F 34. The following code is valid (compiles and runs) in Java:

```
for( ; ; )
    ;
```

T F 35. The output of the following code in Java is:

32

```
int x, answer=1;
for(x=0; x<5; x++) {
    answer = answer * 2;
}
System.out.println( answer);
```

T F 36. The following code prints the words: Aye Done

```
char c='a';
switch (c){
    case 'a': System.out.print("Aye");
    case 'b': System.out.print("Bee");
    case 'c': System.out.print("See");
    break;
}
System.out.println(" Done");
```

II. Multiple Choice (4 points each)

37. Consider the program segment given below. Its output is:

```
for( int i=1; i<=100; i++) {
    System.out.printf("%4d", i);
    if( i%10 == 0) {
        System.out.println();
    }
}
```

- a) All the numbers from 1 to 100, with a line break after the number 10
- b) All the numbers from 1 to 100 in a grid of 10 rows and 10 columns
- c) All the numbers from 1 to 101, 4 per row
- d) All the numbers from 1 to 101, with a line break after every 10 numbers
- e) None of the above

38. Consider the program segment given below. Its output is:

```
String theWord = "MMiissiippii";
for (int i=0; i<theWord.length(); i=i+2) {
    System.out.print(theWord.charAt( i));
}
```

- a) MMiissiippiippii
- b) Mississippi
- c) iippiippiissiMM
- d) MMiissiippiippiippiippiissiMM
- e) None of the above

41. Consider method `first` shown at right. For positive numbers, how would you best describe its return value?

- a) $x + y$
- b) $x * x$
- c) $x * y$
- d) x^y
- e) None of the above

42. Consider method `second` shown at right. For positive numbers, how would you best describe its return value?

- a) $x + y$
- b) $x * x$
- c) $x * y$
- d) x^y
- e) None of the above

```
public int first(int x, int y)
{
    int z=0;
    for (int i=0; i<y; i++) {
        z += 1;
    }
    return z;
}

public int second(int x, int y)
{
    int z=1;
    for (int i=0; i<y; i++) {
        z = first( z, x);
    }
    return z;
}
```

43. Consider the class given below, along with the driver class for it.

<pre>class ClassA { private int x = 1; public void setValue(int val) { x = val + 1; } } //end ClassA</pre>	<pre>class ClassADriver { public void doIt() { int value = 7; ClassA instance1 = new ClassA(); instance1.setValue(5); value = instance1.x; System.out.println("value is: " + value); } } //end ClassADriver</pre>
---	--

When running method `doIt()` in the `ClassADriver` class, the output will be:

- a) value is: 5
- b) value is: 6
- c) value is: 7
- d) doesn't compile
- e) None of the above