CS 102 - Introduction to Programming
Midterm Exam #1 - Prof. Reed
Fall 2008

What is your name?: ___________________________

There are two sections:
I. True/False. . . . . . . . . . . . . . . . . . . . . . 68 points; ( 34 questions, 2 points each)
II. Multiple Choice . . . . . . . . . . . . . . . . . 32 points; ( 8 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 run only in BlueJ. You can’t run Java programs any other way.

For the following 7 statements, tell whether or not each statement is valid.

T F 2. int If = 7;
T F 3. int x = 3.1415;
T F 4. int numberofanswers;
T F 5. char initials = 'DFR';
T F 6. boolean trueORfalse ="true";
T F 7. System.out.println("This class is " + "fun");
T F 8. int average = (int)7.84/3;

T F 9. A constant in Java must also be static, otherwise its value can be changed.

T F 10. An if statement in Java is used to choose between alternatives.

T F 11. Indentation in a Java program can affect which lines do or don’t get executed at a certain point.

T F 12. Code in Java that uses && could be rewritten without &&, using an additional if statement instead.

T F 13. Code in Java that uses || could be rewritten without |, using an additional if statement instead.

if (a && b) // stmt
if (a) // stmt
if (b) // stmt

14. The code shown below is valid. (Valid, but confusing)

```
int x = 3;
x = x + (x * x) / x - x;
```

15. The output of the statement below is: 1

```
System.out.println( -7 % -3);
```

16. After running the code shown below, the value stored in variable y is: 13

```
int y = 7;
y = (-y) + y;
```

17. After running the code shown below, the value stored in variable z is: 15

```
int y = 7;
int z = y + (y++)
```

18. The output of the code shown below is all on one line. Be careful

```
System.out.print("B"); System.out.print("e + " car");
System.out.print("efu + "1");
System.out.print(");
```

19. The output of the code shown below is: o

```
String greeting = "Hello"
System.out.println( greeting.charAt(4) );
```

20. The output of the code shown below is: Same

```
String adieu = "Bye now"
if( adieu == "Buy now"){
    System.out.println("Same");
} else {
    System.out.println("Different");
}
```

21. The code shown below is indented as recommended in class.

```
// assign letter grade based on score
char letterGrade = ' ';
if( score >= 90)
    letterGrade = 'A';
else if( score >= 80)
    letterGrade = 'B';
else if( score >= 70)
    letterGrade = 'C';
else if( score >= 60)
    letterGrade = 'D';
else
    letterGrade = 'F';
```
22. If class Transaction contains a static variable called howMany, then each instance of the class will get its own copy of variable howMany.

23. The code shown below is indented as recommended in class.

```java
// display menu and select menu option
System.out.println("1. Eat lunch.");
System.out.println("2. Watch a movie.");
System.out.println("3. Exit");
System.out.println("Your choice: ");
int menuOption = keyboard.nextInt();

if (menuOption == 1) {
    System.out.println(" Option 1 was chosen.");
} else if (menuOption == 2) {
    System.out.println(" Option 2 was chosen.");
} else if (menuOption == 3) {
    System.out.println(" Option 3 was chosen.");
}
```

24. If the first character of user input is an upper-case vowel ('A', 'E', 'I', 'O', 'U'), then the following code prints out: Upper-case vowel

```java
char theLetter = userInput.charAt(0);
if (theLetter == 'a' || theLetter == 'e' ||
    theLetter == 'A' || theLetter == 'E' ||
    theLetter == 'i' || theLetter == 'o' ||
    theLetter == 'I' || theLetter == 'O' ||
    theLetter == 'u' || theLetter == 'U') &&
    (theLetter >= 'A') && (theLetter <= 'Z') {
    System.out.println("Upper-case vowel");
} else {
    System.out.println("Nope");
}
```

25. The output of the following lines of code is: False is true

```java
boolean False = false;
if (False = true) {
    System.out.println("False is true");
} else {
    System.out.println("False is false");
}
```

26. The output of the program segment below is the text: Small Large

```java
int x = 6; if (x > 3) if (x < 7) 
System.out.print("Small"); else System.out.print("Medium");
System.out.println(" Large");
```

27. The output of the program segment below is: anarchy rules

```java
String jumbo = "anarchy ";
String shrimp = "rules ";
System.out.println(jumbo + shrimp);
```

CS 102 - Intro. to Programming, Midterm Exam #1, page 3 of 8
28. (Careful...) The output of the program segment below is: Exiting

```java
boolean notDone = false;
int numberOfPieces = 4;

if (notDone = true) {
    if (numberOfPieces > 4)
        System.out.println("You won!");
    else {
        System.out.println("Keep playing");
    }
}
System.out.println("Exiting");
```

29. The output of the following statements is:

```java
int x = 3;
int y = 7;
System.out.print( x + y + " ");
System.out.println(" Done");
```

30. The output of the following statements is:

```java
int x = 3;
int y = 7;
System.out.print( "+ " + x + y);
System.out.println(" Done");
```

31. The output of the following statement is:

```java
System.out.println( (int)(2 * 3.5) / 2);
```

32. The following code is valid (compiles and runs) in Java:

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

33. The output of the following code in Java is:

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

34. The following code prints the words:

```java
char c='b';
switch (c){
    case 'a': System.out.print("third");
    case 'b': System.out.print("second");
    case 'c': System.out.print("first");
    break;
}
System.out.println(" Done");
```
II. Multiple Choice (4 points each)

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

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

a) emordnilap
b) emordnila
c) palindrome
d) alindrome
e) None of the above

36. What is the output of the program segment below when an instance of class `Check` is created and used to call method `checkIt()`?

```java
class Check
{
    int x = 0;
    void checkIt()
    {
        int y = 3;
        if( one() )
            if( two(y) )
                x++;
        System.out.println(x);
    }
    boolean one()
    {
        x++;
        return true;
    }
    boolean two( int x)
    {
        x++;
        return true;
    }
}
```//end class Check

a) 0
b) 1
c) 2

d) 3
e) None of the above
37. What is the output of the code given in the two columns below when an object of type `Confuse` is created and used to call method `startUp()`?

```java
class Confuse {
    private int x;
    private int y;

    public Confuse() {
        x = 1; y = 3;
    }

    private void first(int x, int y) {
        x = 4; y = 5;
    }

    private void second(int y) {
        x = y; x++;
    }

    private void third(int s, int t) {
        setXY(s, (x-1));
        s = 3; t = 1;
    }

    private void setXY(int s, int y) {
        x = s;
        this.y = y;
        second(y);
    }

    private void display() {
        System.out.println(x + y);
    }

    public void startUp() {
        first(x, y);
        second(x);
        third(x, y);
        display();
    }
}
```//end class Confuse

![Code snippet]

a) 3
b) 4
c) 5
d) 6
e) None of the above

38. Consider the code given below. If its output is: Date is: F8762 what is the best explanation?

```java
Date startingDate = new Date();
System.out.println("Date is: " + startingDate);
```  

a) The default constructor was not written properly.  
b) Some fields were not initialized, so random values were printed  
c) The copy constructor was not created for class Date  
d) The toString method was not created for class Date  
e) None of the above
39. Assume that you create class Employee that includes an instance of class Date to store the startDate for each employee. Assume that you have written some test code in class EmployeeDriver shown below, where you change the startDate for e1. To your surprise when you run this code the startDate for e2 has changed as well. What is the most likely explanation for this? 1 or 2

```java
class EmployeeDriver {
    public static void main(String[] args) {
        Employee e1 = new Employee;
        Employee e2 = new Employee;
        e1.changeDate(9, 30, 1923);
        System.out.println(e1);
        System.out.println(e2);
    }
}
```

a) The Date class fields are declared as static  
b) The Employee copy constructor does not create a new Date  
c) There is no copy constructor for the Date class  
d) The new value happens to be the same as the default value  
e) None of the above

40. Consider method first shown at right. How would you best describe its return value?

a) \( x + y \)  
b) \( x \times x \)  
c) \( x \times y \)  
d) \( x^y \)  
e) None of the above

41. Consider method second shown at right. How would you best describe its return value?

a) \( x + y \)  
b) \( x \times x \)  
c) \( x \times y \)  
d) \( x^y \)  
e) None of the above

40. Consider method first shown at right. How would you best describe its return value?

a) \( x + y \)  
b) \( x \times x \)  
c) \( x \times y \)  
d) \( x^y \)  
e) None of the above

41. Consider method second shown at right. How would you best describe its return value?

a) \( x + y \)  
b) \( x \times x \)  
c) \( x \times y \)  
d) \( x^y \)  
e) None of the above
42. Consider the class given below, along with the driver class for it.

```java
class ClassA {
    private int x = 1;
    public ClassA(int x) {
        this.x = x + 1;
    }
}
//end ClassA

class ClassADriver {
    ClassADriver() {
        int value = 7;
        ClassA instance1 = new ClassA(value);
        value = instance1.x;
        System.out.println("value is: "+value);
    }
}
//end ClassADriver
```

When creating an instance of ClassADriver, the output will be:

a) value is: 7
b) value is: 8
c) value is: 9
d) doesn’t compile
e) None of the above

Answer would be 8, except that x is private, which means it does not compile.