

# CSC 116 – 002 Exam 1 Spring 2005

Name: \_\_\_\_\_

Lab # and/or Lab Instructor: \_\_\_\_\_ (1 point)

This exam is worth 100 points. You may use your personal copy of the Wu textbook and your personal notes. You may not use any other materials, including your neighbor. You may not share your textbook, notes, or calculators.

---

## Part 1: Vocabulary (20 points)

- |                           |  |
|---------------------------|--|
| <b>Syntax</b>             | 1. Structure or pattern of a programming language.   |
| <b>Instance variables</b> | 2. Used to define the states or data of an object.   |
| <b>Class</b>              | 3. A template of something you are trying to represent.  |
| <b>Constructor</b>        | 4. Initializes an object when it is created.   |
| <b>Source code</b>        | 5. Human readable code.  |
| <b>Methods</b>            | 6. Used to define actions or behaviors of an object.   |
| <b>Integers or ints</b>   | 7. A primitive data type for numbers that does not have a fractional part.   |
| <b>Object</b>             | 8. An instance of a class  |
| <b>Keyword</b>            | 9. A word that has special meaning in a programming language and cannot be used by programmers as a name for classes, methods, or variables. |
| <b>Compiler</b>           | 10. Translates source code into machine code   |

## Part 2: Primitive Data Types (20 points)

### Arithmetic Operations (5 points)

You have three variables listed below. Compute the results of using these values in the following expressions. The values of these variables will not change through the computation of the following expressions (a will always have the value 42).

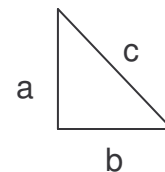
```
int a = 42;  
int b = 7;  
double c = 4.0;
```

1.  $a + b = 49$
  2.  $b * c = 28.0$
  3.  $a / c = 10.5$
  4.  $a / b = 6$
  5.  $a \% b = 0$
- 

### Write Your Own Expressions

You are given two variables that contain the two short sides of a right triangle. Assume they have been initialized to some appropriate value.

```
double sideA;  
double sideB;
```



Provide Java statement(s) to calculate and store the area of the triangle. Remember that the area of a triangle =  $(1/2) * \text{base} * \text{height}$  and that this is a right triangle. Also remember to declare the variable that you will store the area in. (5 points)

```
double area = 0.5 * sideA * sideB;
```

Provide Java statement(s) to calculate and store the length of the 3<sup>rd</sup> side of the triangle. Remember that the Pythagorean Theorem is  $a^2 + b^2 = c^2$ . (5 points)

```
double sideC = Math.sqrt(sideA * sideA + sideB * sideB);  
double sideC = Math.sqrt(Math.pow(sideA,2) + Math.pow(sideB,2));
```

## Booleans (5 points)

Assume that variables `day` and `month` contain the current day and month. Complete the method to return true if this is your birthday (you may use literal numbers to test the values in your expression and you will return the answer to a Boolean expression).

For example, if your birthday is October 14<sup>th</sup>, then you want to know if the month is 10 and the day is 14.

```
public boolean isMyBirthday (int day, int month) {  
  
    return day == 14 && month == 10; //or their b-day (valid!)  
}
```

## Part 3: Strings (10 points)

### String Comparisons (5 points)

Assume that variables `student1` and `student2` contain the names of students in a class. Complete the method to return true if the names of the students are the same.

```
public boolean isSameStudent (String student1, student2) {  
  
    return student1.equals(student2);  
}
```

### String Methods (5 points)

Given variable `s`:

```
String s = "The quick brown fox jumps over the lazy dog.";
```

Create the Java statement that will store the string "rown fox jum" into the `subString` variable.

```
String subString = s.substring(11,23);
```

Create the Java statement to calculate the length of the `s` variable.

```
int length = s.length();
```

## Part 4: Classes (30 points)

### Writing a Class

You want to write a class to help a non-profit organizations keep track of the number of donations they have received. The NonProfitOrg class knows the name of the organization, the total amount of money donated to the organization, and the number of donors (both companies and individuals).

```
public class NonProfitOrg {  
  
    /**  
     * Instance variables for the NonProfitOrg class that store  
     * store the name of the organization, the total amount of  
     * money donated to the organization, and the number of  
     * donors. Please make sure to specify the correct access  
     * identifier.  
     */  
  
    private String orgName;  
    private double money;  
    private int numDonors;  
  
  
  
  
    /**  
     * Constructor for the NonProfitOrg class that provides  
     * initial values for each of the instance variables. This  
     * is a complete constructor.  
     */  
  
    public NonProfitOrg(String name, double money, int numD) {  
        this.orgName = name;  
        this.money = money;  
        this.numDonors = numD;  
        //Make sure to watch for this being used if the param  
        //and instance variable have the same name  
    }  
}
```

```

/**
 * This method accepts a new donation. The money from the
 * donations is added to the total amount of donated money
 * and the number of donors is incremented by 1
 */

public void acceptDonation(double money) {

    this.money += money;
    numDonors++;

    //This method may have a parameter increasing the
    //number of donors as well - anything that
    //looks right

}

/**
 * This method returns the average amount of money donated
 * across all donors. The average may contain a fraction
 * part.
 */

public double averageDonations() {

    double average = this.money / this.numDonors;
    return average;

    //If parameters are used correctly throughout the
    //whole example class then no points taken off

}

```

```

/**
 * Below is a main method. Create two NonProfitOrg objs.
 * You can provide names, total donations thus far, and the
 * number of donors so far. For one of the NonProfitOrg
 * objects, accept a new donation. For the other object,
 * print (to standard out) the average amount of donations
 * received.
 */
public static void main(String [] args) {

    //Create two NonProfitOrg objects
    NonProfitOrg redCross = new NonProfitOrg("RedCross",
        300.00, 5);

    NonProfitOrg peaceCore = new NonProfitOrg("Peace
        Core", 5000.00, 10);


    //Accept a new donation for one of the objects
    redCross.acceptDonation(500.00);


    //Print the average amount of donations for the other
    //object
    System.out.println(peaceCore.averageDonations());


} //End main
} //End class

```

## Part 5: Conditionals and Loops (20 points)

### While Loops (5 points)

Write a while loop to calculate  $3^7$ .

```
int power = 1;
int counter = 1;
while(counter <= 7) {

    power *= 3;

    counter++;

}
```

### For Loops (5 points)

Write a for loop to calculate  $3^7$ .

```
int power = 1;
for(int i = 1; i <= 7; i++) {

    power *= 3;

}
```

### Nested Loops (10 points)

Write a nested loop that produces the following output:

```
1
12
123
1234
```

```
for(int i = 1; i <= 4; i++) {

    for(int j = 1; j <= i; j++) {

        System.out.print(j);

    }

    System.out.println("");

}
```