

UNIVERSITY OF WISCONSIN-LA CROSSE
Department of Computer Science

CS 120
Midterm Exam 02

Software Design I

Fall 2016
15 November 2016

- Do not turn the page until instructed to do so.
- This booklet contains 8 pages including the cover page.
- This is a closed-book exam. All you need is the exam and a writing utensil.
- You have exactly 55 minutes.
- The maximum possible is 55.

PROBLEM	SCORE
1	
2	
3	
4	
5	
TOTAL	

NAME: _____

1. (10 pts.) **TRUE/FALSE.**

For each of the following, indicate whether the statement is true or false.

You do not need to explain your answers.

- a. A **private** class method can only be called by other **private** methods in that same class.
- b. A **private** instance variable can only be used in **private** methods in that same class.
- c. A **public** class method can be called from within **public** methods in other classes.
- d. A **public** class method can be called from within **private** methods in other classes.
- e. A **void** method can have a **return** statement.

- f. Suppose that **methodA()** returns an **int** value. Then the following is a legal line of code:

```
double num2 = methodA() * 2;
```

- g. Suppose that **methodB()** returns a **double** value. Then the following is a legal line of code:

```
int num3 = (int) methodB() * 2.0;
```

- h. Suppose that **methodA()** and **methodB()** are part of the same class; furthermore, suppose that **methodA()** returns an **int** value, and that **methodB()** takes two **int** values as input. Then the following is a legal line of code:

```
methodB( methodA(), methodA() );
```

- i. Assuming integer variables **i** and **j** are properly declared and instantiated, the following code will swap their values:

```
int temp = i;  
j = temp;  
i = j;
```

- j. The following contains an error and will not compile:

```
for ( int i = 0; i < 5; i++ )  
{  
    System.out.println( i );  
}  
System.out.println( i );
```

2. (10 pts.) **SHORT ANSWER.**

a. (2 pts.) How many times will these loops run (assuming they are in a correct program)?

```
(1)      int num = 10;
         while ( num >= 0 )
         {
             System.out.println( num );
             num--;
         }
```

Answer: _____

```
(2)      for ( int i = 0; i <= 8; i = i * 2 )
         {
             System.out.println( i );
         }
```

Answer: _____

b. (4 pts.) Java uses *pass-by-value* semantics for method calls. For inputs of *primitive type*, the method uses a copy of the parameter's _____.

For *reference types*, the method uses a copy of the parameter's _____.

Thus, a method *can* use the input parameter to change things about the original value of the variable used when calling the method if the input is of _____ type,

but *can not* change the original if it is of _____ type.

c. (4 pts.) Suppose a class, **Driver**, contains the following method call on a **Helper** object:

```
Helper help = new Helper();
String s = "test";
int i = help.count( s, s.charAt( 0 ) );
```

What will be the method declaration (i.e., the first line) of method `count()` be?

3. (10 pts.) **CODE EVALUATION.**

- a. (3 pts.) Write out what will be returned by the following method, for the method calls given below:

```
private boolean process( String s1, String s2 )
{
    for ( int i = 0; i < s1.length(); i++ )
    {
        boolean b = false;
        for ( int j = 0; j < s2.length(); j++ )
        {
            if ( s1.charAt( i ) == s2.charAt( j ) )
            {
                b = true;
            }
        }
        if ( !b )
        {
            return false;
        }
    }
    return true;
}
```

i. `process("Dogs", "Doges");` _____

ii. `process("Doges", "Dogs");` _____

iii. `process("", "Hello");` _____

- b. (2 pts.) In general, what does the previous method check when given two inputs, **s1** and **s2**? (That is, describe briefly what conditions make the method return a **true** value, and what conditions make it return a **false** value.)

c. (5 pts.) Write out what is printed by the following method when **both** inputs are 4.

```
private void printNums( int num1, int num2 )
{
    for ( int i = 1; i <= num1; i++ )
    {
        System.out.print( i + "." );
        for ( int j = 1; j <= num2; j++ )
        {
            int d = i / j;
            System.out.print( " " + d );
        }
        System.out.println();
    }
}
```

4. (10 pts.) CODING NESTED LOOPS

Fill in the `main()` method in the class below so that when it runs it prints output (using `System.out.println()` or `print()`) that looks like this:

```
1 + 2 + 3 + 4 + 5 = 15
2 + 3 + 4 + 5 + 6 = 20
3 + 4 + 5 + 6 + 7 = 25
4 + 5 + 6 + 7 + 8 = 30
5 + 6 + 7 + 8 + 9 = 35
```

For full points, your code must use a pair of **nested loops**, each of which is actually used to generate the output. (You may use whatever types of loops you choose.)

Hint: use the loops to actually calculate the sums that are displayed, as well as using them to produce the printed output.

```
public class Main
{
    public static void main( String[] args )
    {

    }
}
```

5. (15 pts.) **CODE COMPLETION.**

On the next page, complete the given **Main** class as follows:

- a. Write the method **countLetters()** so that it works with the code as given:
 - i. This method will take a **String** as input.
 - ii. It will return the number of letters ('a' ... 'z' or 'A' ... 'Z') in the input.
- b. Write the method **reverse()** so that it works with the code as given:
 - i. This method will take a **String** as input.
 - ii. It will return as output a new **String** that is the reverse of the input.
- c. Write the method **shuffle()** so that it works with the code as given:
 - i. This method will take in two **String** inputs.
 - ii. It will return a new **String** as output.
 - iii. If the two inputs are the same length, the output is the result of shuffling the characters of the two inputs together, one at a time, starting with the first character of the first input, then the first character of the second input, followed by the second character of the first input, then the second character of the second input, and so on.
 - iv. If the two inputs are different lengths, then it will return "NO SHUFFLE POSSIBLE".

When complete, the code should produce the following output when run.

```
Hello!! -> 5
!!olleH -> 5
aBaBaBaBaB
NO SHUFFLE POSSIBLE
NO SHUFFLE POSSIBLE
```

```
public class Main
{
    public static void main( String[] args )
    {
        Main m = new Main();
        String str1 = "Hello!!";
        System.out.println( str1 + " -> " + m.countLetters( str1 ) );
        String str2 = m.reverse( str1 );
        System.out.println( str2 + " -> " + m.countLetters( str2 ) );
        System.out.println( m.shuffle( "aaaaa", "BBBBB" ) );
        System.out.println( m.shuffle( "start", "END" ) );
        System.out.println( m.shuffle( "END", "start" ) );
    }
}
```

```
\\ Complete Main code for methods here (Question 5)
```

```
}
```