UNIVERSITY OF WISCONSIN-LA CROSSE
Department of Computer Science

CS 120 Software Design I Fall 2017
Midterm Exam 02 Monday, 13 November 2017

- **Do not turn the page until instructed to do so.**
- **This booklet contains 7 pages including the cover page.**
- **This is a closed-book exam. All you need is the exam and a writing utensil. (A calculator is permitted.)**
- **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 `for` loop must always increase or decrease an integer counter variable by `1` each time.

   b. The following code contains a compile-time error (i.e., will not compile properly):

```
for ( int i = 0; i < 10; i++ ) {
    System.out.println( i );
}
System.out.println( "Final value of i = " + i );
```

   c. The following code contains a compile-time error (i.e., will not compile properly):

```
int i = 0;
String s = "Test";
while ( i <= s.length() ) {
    System.out.println( s.charAt( i ) );
}
```

   d. A `private` class method can only be called by `main()`, but no other methods in that class.

   e. A `private` class method can only be called by other `private` methods in that class.

   f. A `public` class method can be called by `public` **and** `private` methods in other classes.

   g. A `void` method can still have a `return` statement.

   h. Suppose that `methodA()` returns a `double`. Then the following is a legal line of code:

```
int num = (int) methodA() / 5;
```

   i. Suppose that `methodB()` returns an `int`. Then the following is a legal line of code:

```
double num2 = methodB();
```

   j. Suppose that `methodC()` returns a `String`. Then the following is a legal line of code:

```
int num3 = methodC().length();
```

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 = 5;
while ( num <= 0 )
{
    System.out.println( num );
    num++;
}
```

Answer: _____

(2)
```
for ( int i = 1; i <= 10; i = i * 2 ) {
    System.out.println( i );
}
```

Answer: _____

b. (4 pts.) Consider the following method, with ***one of*** the two different versions of code inside it:

```
private int checkNums( int i, int j )
{
    if ( i < j )                    |    if ( j > i )
        return i;                   |        return i;
    else if ( i > j )               |    else
        return j;                   |        return j;
}
```

Answer True or False:

(i) The first (left-hand) version of the code will compile: _____

(ii) The second (right-hand) version of the code will compile: _____

If you answered **True** to **both** questions (so both versions compile), then explain what each version of the method will do. If you answered **False** for either version, then explain why the code will fail to compile:

_____

_____

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

```
Helper h = new Helper();
double m = 1;
int num = h.calculate( m, m * 2, m / 2 );
```

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

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

a. Write out what will be **returned** by the following method, for each call given:

```
private String process( String s1, String s2 )
{
    int len = s1.length();
    if ( s2.length() < len ) {
        len = s2.length();
    }

    String out = new String();
    for ( int i = 0; i < len; i++ ) {
        if ( s1.charAt( i ) == s2.charAt( i ) ) {
            out = out + s1.charAt( i );
        }
    }
    return out;
}
```

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

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

iii. `process( "Bing", "Bong" )` _____

iv. `process( "Apple", "Banana" )` _____

v. `process( "", "Hello" )` _____

b. Write out what is printed by the following method, given inputs `4` and `5`, in that order:

```
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 below to print output (using `System.out.println()` or `print()`):

```
 1 =   1/1
 2 =   2/1  4/2
 3 =   3/1  6/2  9/3
 4 =   4/1  8/2 12/3 16/4
 5 =   5/1 10/2 15/3 20/4 25/5
 6 =   6/1 12/2 18/3 24/4 30/5 36/6
 7 =   7/1 14/2 21/3 28/4 35/5 42/6 49/7
 8 =   8/1 16/2 24/3 32/4 40/5 48/6 56/7 64/8
 9 =   9/1 18/2 27/3 36/4 45/5 54/6 63/7 72/8 81/9
10 =  10/1 20/2 30/3 40/4 50/5 60/6 70/7 80/8 90/9 100/10
```

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.)

Note: ***spacing matters***—for full points, your code should print out results that are lined up properly in columns exactly as shown above.

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









}
```

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

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

a. Write the method `getChar()` so that it works with the code as given:
   i. The method will take a `String` and an `int` as input.
   ii. It will return a `char`, equal to the character found at the integer position given.
   iii. If the `String` is **empty** (and has no characters), or the integer position given is **too small** or **too large**, a single period is returned (`'.'`).

b. Write the method `countChar()` so that it works with the code as given:
   i. This method will take a `String` and a `char` as input.
   ii. It will return the count of how many times the character occurs in the `String`.

c. Write the method `removeAll()` so that it works with the code as given:
   i. This method will take a `String` and a `char` as input.
   ii. It will return a new `String` as output, consisting of the input `String`, with every occurrence of the input character removed.

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

```
. s .
3 0
thisisatest
thiiatet
```

```
public class Q5
{
    public static void main( String[] args )
    {
        Q5 q = new Q5();
        char ch = q.getChar( "Test", -11 );
        System.out.print( ch + " " );
        ch = q.getChar( "Test", 2 );
        System.out.print( ch + " " );
        ch = q.getChar( "", 0 );
        System.out.println( ch );
        int i = q.countChar( "this is a test", 't' );
        System.out.print( i + " " );
        i = q.countChar( "THIS IS A TEST", 't' );
        System.out.println( i );
        String str = q.removeAll( "this is a test", ' ' );
        System.out.println( str );
        str = q.removeAll( str, 's' );
        System.out.println( str );
    }
```

```
    \\ complete code for methods here (and on back of page)

}
```