COS 126	General Computer Science	Fall 2017

Written Exam 1

This exam has 7 questions (including question 0) worth a total of 70 points. You have 50 minutes. Write all answers inside the designated spaces.

Policies. The exam is closed book, except that you are allowed to use a one page cheatsheet (8.5-by-11 paper, one side, in your own handwriting). No electronic devices are permitted.

Discussing this exam. Discussing the contents of this exam before solutions have been posted is a violation of the Honor Code.

This exam. Do not remove this exam from this room. Write your name, NetID, and the room in which you are taking the exam in the space below. Mark your precept number. Also, write and sign the Honor Code pledge. You may fill in this information now.

Name:

NetID:

Exam Room:

Precept:

"I pledge my honor that I will not violate the Honor Code during this examination."

Signature

0	1	2	3	4	5	6	total

0. **Miscellaneous. (1 point)**

   (a) Write your name, NetID, Exam Room, and Precept Number in the space provided
       on the front of this exam.

   (b) Write and sign the Honor Code pledge on the front of this exam.

1. **Java expressions. (14 points)**

   What is the value of each of the following Java expressions? To express your answer,
   write a Java literal of the appropriate type, such as 0, 0.0, false, or "0". If an
   expression results in a compile-time or run-time error, write ERROR for its value. Assume
   that the variables x, y, and z have been initialized as follows:

   ```
   int x = 3;
   int y = (int) 9.9;
   double z = (double) 2;
   ```

Java expression	*value*
x	3
y	
z	
"2" + "2"	
"4" + x	
y / z	
y % x	
(double) "4"	
(double) (y / x)	
x != y && true	
false && false || true	

2. **Shell Basics. (7 points)**

```
public class P {
    public static void main(String[] args) {
        int num = StdIn.readInt();
        StdOut.println(num / 2);
    }
}
```

Assume the following:

(a) The file "in.txt" has a single number 12 in it; and

(b) if keyboard input is needed, you will type the number 6.

Complete the table. If the command results in an error, write "yes" in the "error" column and leave everything else blank. If not, write "no," and complete the rest. If the command requires keyboard input, write "yes" in the "keyboard input" column. If not, write "no". If the command produces terminal output, write "yes" in the "terminal output" column. If not, write "no". If the command produces an output file called out.txt, write "yes" in the "out.txt file" column. If not, write "no". Write the result of the command in the "output text" column regardless of destination.

	command	error	keyboard input	terminal output	out.txt file	output text
0	java-introcs P	no	yes	yes	no	3
1	in.txt < java-introcs P					
2	in.txt | java-introcs P					
3	java-introcs P < in.txt					
4	java-introcs P | out.txt					
5	java-introcs P > out.txt					
6	java-introcs P < in.txt > out.txt					
7	java-introcs P < in.txt | P					

3. **Debugging. (12 points)**

Consider the following program. The author intended for it to print the square of `n` numbers (zero through `n - 1`) and to print "7 IS MY FAVORITE" only when `n` is 7. The program is properly invoked with a single integer argument (`n`) on the command line.

```
    public static void main(String[] args) {
1        int n = Integer.parseInt(args[1]);
         int i;
2        for (i = 0; i < n; i++);
3          printf("i * i = %d\n", i * i);
4        if (n = 7)
5          StdOut.printf("%d IS MY FAVORITE\n");
    }
```

Each numbered line has a bug. (The numbers are not part of the program.) Using the table below, for each line, write "C" if the bug is a compile-time error and "R" if it is a run-time error (when properly invoked). Also, for each line, write the corrected version. One space has been filled in for you.

Line	"C" or "R"	Correction
1		
2		
3		
4		
5	R	

4. **Loops and Conditionals. (12 points)**

```
public static void main(String[] args) {
    int n = Integer.parseInt(args[0]);
    int i = 2;

    while (n > 1) {
        if (n % i == 0) {
            StdOut.print(i + " ");
            n = n / i;
        } else {
            i++;
        }
    }
}
```

(a) What does this program print if n is 1000? Draw a box around your answer.

(b) Explain this program's function in one English sentence. Do not paraphrase the code.

5. **Recursion. (12 points)**

Consider the following mutually recursive functions.

```
public static int mystery1(int n) {
    if (n == 0) return 0;
    else return mystery2(n - 1);
}

public static int mystery2(int n) {
    if (n == 0) return 1;
    else return mystery1(n - 1);
}
```

(a) Complete the following table with the value returned by the function call.

n	mystery1(n)	mystery2(n)
0		
1		
2		
3		
4		
5		

(b) Given a positive integer `n`, what does the function call `mystery1(n)` return? Answer this by explaining its function in one English sentence of no more than 15 words. Do not paraphrase the code.

(c) Write a new function `mystery()` that performs the same function as `mystery1()` in one line of code. Fill in the blank.

```
public static int mystery(int n) (


    return _____;
}
```

6. **Arrays. (12 points)**

   Consider the following program (the numbers on the left are line numbers for reference only and are not part of the code):

```
1     public static void main(String[] args) {
2          int N = 10;
3          int UNKNOWN = 2;
4          int P = 1;
5          int NONP = 0;
6
7          int[] a = new int[N];
8
9          for (int j = 2; j < N; j++)
10             a[j] = UNKNOWN;
11
12         for (int i = 2; i < N; i++) {
13             if (a[i] == UNKNOWN) {
14                 a[i] = P;
15                 StdOut.print(i + " ");
16                 for(int j = i; i * j < N; j++)
17                     a[i * j] = NONP;
18             }
19         }
20     }
```

(a) What will the contents of the `a[]` array be after the loop at lines 9-10 is complete? Draw a box around your answer.

(b) What will the program print? Draw a box around your answer.

Name: _____ **NetID:** _____ **Precept:** _____