COS 126	General Computer Science	Spring 2016

Written Exam 1

This exam has 8 questions (including question 0) worth a total of 70 points. You have 50 minutes.

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 calculators or other electronic devices are permitted. *This exam is preprocessed by computer. If you use pencil (and eraser), write darkly. Write all answers inside the designated rectangles. Do not write on corner marks.*

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. *Print your name, NetID, and precept in the space below. Write and sign the Honor Code pledge.*

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

   (a) Write your name and Princeton NetID in the space provided on the front of the exam, and circle your precept number.

   (b) Write and sign the honor code on the front of the exam.

1. **Java basics. (10 points)**

   (a) Give 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 = 111;
int y = 222;
int z = 333;
```

Java expression	*value*
`x`	`111`
`x - 2.0 * y + z`	
`x / (z - x - y)`	
`Math.sqrt(x / (x + z))`	
`x + "222" + (y + z)`	
`(x <= y <= z)`	
`!((x <= 2*y) && (y <= 2*x))`	

(b) Suppose that the variables `a`, `b`, and `c` are initialized as follows.

```
int a = 11111;
int b = 22222;
int c = 0;
```

Consider the following statements.

1. `a = b;`
2. `a = c;`
3. `c = a;`
4. `c = b;`
5. `b = a;`
6. `b = c;`

For example, `1 3 6` refers to the following sequence of statements:

```
a = b;
c = a;
b = c;
```

Which of the following sequences of statements will swap the values of `a` and `b`, i.e., leave the value `22222` in `a` and `11111` in `b`? Mark all that apply.

`1 3 5`	`2 3 5`	`3 1 5`	`4 1 5`
`1 3 6`	`2 3 6`	`3 1 6`	`4 1 6`
`1 4 5`	`2 4 5`	`3 2 5`	`4 2 5`
`1 4 6`	`2 4 6`	`3 2 6`	`4 2 6`
`1 5 3`	`2 5 3`	`3 5 1`	`4 5 1`
`1 5 4`	`2 5 4`	`3 5 2`	`4 5 2`
`1 6 3`	`2 6 3`	`3 6 1`	`4 6 1`
`1 6 4`	`2 6 4`	`3 6 2`	`4 6 2`

2. **Properties of arrays and functions. (10 points)**

(a) Which of the following statements are true for *Java arrays*. Assume that the arrays are one-dimensional and integer-valued. Mark all that apply.

i. The statement `int[] a = new int[n];` declares and creates a new array of length `n`, with each element initialized to `0`.

ii. You can refer to the first element in an array `a[]` with `a[0]` and the last element with `a[a.length]`.

iii. Each element `a[i]` in an array `a[]` can be used like an ordinary variable of type `int`: as a term in an expression or as the left-hand side in an assignment statement.

iv. If `a[]` and `b[]` are arrays of the same length, then the expression `a + b` evaluates to a new array whose elements are the sum of the corresponding elements in `a[]` and `b[]`.

v. The values of the elements in an array are stored consecutively in the computer's memory.

(b) Which of the following statements are true for *Java functions* (static methods). Mark all that apply.

i. A `.java` file can contain the definition of more than one function, but each function must have a different name.

ii. Two different functions in the same `.java` file can declare local variables with the same name.

iii. A function must contain exactly one `return` statement (unless its return type is `void`).

iv. A function may either return a value to the caller or produce a side effect (such as consuming input or producing output), but cannot do both.

v. The scope of a local variable declared within a function is limited to that function.

3. **Loops and conditionals. (10 points)**

(a) Consider the following Java code fragment.

```
int N = 100;
for (int i = 0; i < N; i++) {
    for (int j = 0; (j < N) && (j != i); j++) {
        System.out.println(i + "-" + j);
    }
}
```

Which of the following will appear on standard output? Mark all that apply.

`0-0` `0-1` `1-0` `8-8` `2-98` `98-2` `100-99`

(b) Consider the following Java code fragment. Assume that `x`, `y`, and `z` are variables of type `int` and initialized to `10`, `5`, and `0`, respectively.

```
if (x >= y) {
    System.out.println("A");
    if (x >= z) {
        System.out.println("B");
    }
    else {
        System.out.println("C");
    }
}
if ((x < y) && (x < z)) {
    System.out.println("D");
}
else {
    System.out.println("E");
    if (y >= z) {
        System.out.println("F");
    }
}
```

Which of the following will appear on standard output? Mark all that apply.

`A` `B` `C` `D` `E` `F` `G`

4. **Arrays. (10 points)**

Consider the following Java code fragment.

```
int[] a = { 1, 6, 5, 3, 0, 2, 4 };
int n = a.length;

int[] b = new int[n];
for (int i = 0; i < n; i++)
    b[a[i]] = i;

int[] c = new int[n];
for (int i = 0; i < n; i++)
    c[i] = a[b[i]];
```

What are the values of the elements in the arrays `b[]` and `c[]` after the above code fragment is executed? Write your answers in the space below.

a[]	1	6	5	3	0	2	4

b[]							

c[]							

5. **Standard input, standard output, and redirection. (10 points)**
Consider the following Java program.

```
public class Mystery {
    public static void main(String[] args) {
        int current = -1;
        int count = 0;
        while (!StdIn.isEmpty()) {
            int x = StdIn.readInt();
            if (x != current) {
                if (count > 0) StdOut.print(count);
                current = x;
                count = 0;
            }
            count++;
        }
    }
}
```

(a) Assume the contents of the file `input.txt` are given below.

```
% more input1.txt
0 0 0 0 1 1 1 1
```

Suppose that you execute the following command. What is printed on standard output? Write your answer in the space provided.

```
% java-introcs Mystery < input1.txt
```

(b) Repeat the previous question, but with the following input file.

```
% more input2.txt
0 0 0 0 1 1 1 1 0 0 0 0 1
1 1 0 0 0 1 1 1 0 1 1 1
```

```
% java-introcs Mystery < input2.txt
```

6. **Functions. (10 points)**

The `majority()` function takes three boolean arguments and returns `true` if two (or more) of the arguments are `true`; and `false` otherwise. Complete *two* implementations of `majority()` below by filling in the letter of one of the expressions below in each provided space. You may use each letter once, more than once, or not at all (but you may not use any other code).

A. false	F. x && y	J. x || y	N. count++
B. true	G. x && z	K. x || z	O. count--
C. x	H. y && z	L. y || z	P. count
D. y	I. x && y && z	M. x || y || z	Q. count <= 1
E. z			R. count >= 2

(a)

```
public static boolean majority(boolean x, boolean y, boolean z) {

    int count = 0;

    if (_____) _____;

    if (_____) _____;

    if (_____) _____;

    return _____;
}
```

(b)

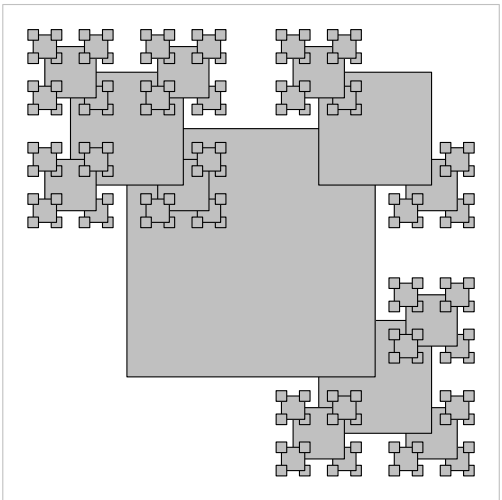
```
public static boolean majority(boolean x, boolean y, boolean z) {

    if      (_____) return _____;

    else if (_____) return _____;

    else if (_____) return _____;

    else            return _____;
}
```

7. **Recursive graphics. (9 points)**

Design a recursive function with the signature

```
public static void draw(int n, double x, double y, double length)
```

so that the call `draw(5, 0.5, 0.5, 0.5)` produces the following *intermediate* result after drawing the 214^{th} shaded square.



The six statements in the function body are given below, but not necessarily in order.

```
1    if (n == 0) return;
2    drawShadedSquare(x, y, length);
3    draw(n-1, x - length/2, y + length/2, length/2.2);   // upper left
4    draw(n-1, x + length/2, y + length/2, length/2.2);   // upper right
5    draw(n-1, x - length/2, y - length/2, length/2.2);   // lower left
6    draw(n-1, x + length/2, y - length/2, length/2.2);   // lower right
```

The helper function `drawShadedSquare()` draws a gray square of the specified side length, outlined in black, and centered at `(x, y)`.

Which of the following must be true for any possible ordering of the six statements that produces the intermediate result shown above? Mark all that apply.

(a) Statement 1 appears first.

(b) Statement 2 appears after statement 5.

(c) Statement 2 appears before statements 3, 4, 5, and 6.

(d) Statements 3 and 6 appear before statements 4 and 5.

(e) If statements 3 and 6 are swapped, the function will still produce the same intermediate result.