

Midterm 1 Written Exam

There are seven questions on this exam, weighted as indicated at the top of each page. The questions are *not in order of difficulty*. If a question seems difficult to you, skip it and come back to it.

Policies. The exam is closed book, though you are allowed to use a single-page two-sided hand-written cheatsheet. No calculators or other electronic devices are permitted. Give your answers and show your work in the space provided. You have 50 minutes to complete the test. **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.

This page. *Print your name, NetID; fill in your precept on this page;* write and sign the Honor Code pledge.

Discussing this exam. As you know, discussing the contents of this exam before solutions have been posted is a serious violation of the Honor Code.

“I pledge my honor that I have not violated the Honor Code during this examination.”

NONE

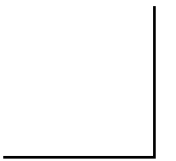
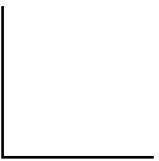
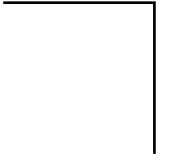
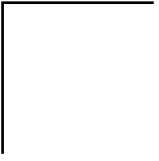
Name:

NetID:

Room:

Precept:

P01	P01A	P01B	P01C	P02	P02A	P02B	P04	P05	P06	P06A	P07	P08	
<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	
									P11	P11A	P12	P13	P14
									<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>



Q1. Java Basic (10 points).

A. Assume that variables `x`, `y`, and `z` have been initialized as follows:

```
int x = 111;  
int y = 222;  
int z = 333;
```

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.

<i>Java expression</i>	<i>value</i>
<code>x</code>	111
<code>x - 2.0 * y + z</code>	<input type="text"/>
<code>x / (z - x - y)</code>	<input type="text"/>
<code>Math.sqrt(x / (x + z))</code>	<input type="text"/>
<code>x + "222" + (y + z)</code>	<input type="text"/>
<code>(x <= y <= z)</code>	<input type="text"/>
<code>!((x <= 2*y) && (y <= 2*x))</code>	<input type="text"/>

B. Suppose that the variables a, b, and c are initialized as follows:

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

And consider the following statements.

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

Thus a sequence written 1 3 5 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 <input type="checkbox"/>	2 3 5 <input type="checkbox"/>	3 1 5 <input type="checkbox"/>	4 1 5 <input type="checkbox"/>
1 3 6 <input type="checkbox"/>	2 3 6 <input type="checkbox"/>	3 1 6 <input type="checkbox"/>	4 1 6 <input type="checkbox"/>
1 4 5 <input type="checkbox"/>	2 4 5 <input type="checkbox"/>	3 2 5 <input type="checkbox"/>	4 2 5 <input type="checkbox"/>
1 4 6 <input type="checkbox"/>	2 4 6 <input type="checkbox"/>	3 2 6 <input type="checkbox"/>	4 2 6 <input type="checkbox"/>
1 5 3 <input type="checkbox"/>	2 5 3 <input type="checkbox"/>	3 5 1 <input type="checkbox"/>	4 5 1 <input type="checkbox"/>
1 5 4 <input type="checkbox"/>	2 5 4 <input type="checkbox"/>	3 5 2 <input type="checkbox"/>	4 5 2 <input type="checkbox"/>
1 6 3 <input type="checkbox"/>	2 6 3 <input type="checkbox"/>	3 6 1 <input type="checkbox"/>	4 6 1 <input type="checkbox"/>
1 6 4 <input type="checkbox"/>	2 6 4 <input type="checkbox"/>	3 6 2 <input type="checkbox"/>	4 6 2 <input type="checkbox"/>

Q2. 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.

	True	False
The statement <code>int[] a = new int[n];</code> declares and creates a new array of length <code>n</code> , with each element initialized to 0.	<input type="radio"/>	<input type="radio"/>
You can refer to the first element in an array <code>a[]</code> with <code>a[0]</code> and the last element with <code>a[a.length]</code> .	<input type="radio"/>	<input type="radio"/>
Each element <code>a[i]</code> in an array <code>a[]</code> can be used like an ordinary variable of type <code>int</code> : as a term in an expression or as the left-hand side in an assignment statement.	<input type="radio"/>	<input type="radio"/>
If <code>a[]</code> and <code>b[]</code> are arrays of the same length, then the expression <code>a + b</code> evaluates to a new array whose elements are the sum of the corresponding elements in <code>a[]</code> and <code>b[]</code> .	<input type="radio"/>	<input type="radio"/>
The values of the elements in an array are stored consecutively in the computer's memory.	<input type="radio"/>	<input type="radio"/>

B. Which of the following statements are true for *Java functions* (static methods)?

	True	False
A <code>.java</code> file can contain the definition of more than one function, but each function must have a different name.	<input type="radio"/>	<input type="radio"/>
Two different functions in the same <code>.java</code> file can declare local variables with the same name.	<input type="radio"/>	<input type="radio"/>
A function must contain exactly one <code>return</code> statement (unless its return type is <code>void</code>).	<input type="radio"/>	<input type="radio"/>
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.	<input type="radio"/>	<input type="radio"/>
The scope of a local variable declared within a function is limited to that function.	<input type="radio"/>	<input type="radio"/>

Q4. Arrays (10 points).

A. 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[]							
-----	--	--	--	--	--	--	--

Q5. Standard Input, Standard Output, and Redirection (10 points).

Consider the following Java class contained in `Mystery.java`, and compiled:

```
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 (note: the line starting with `%` is what *you* type on the command line prompt / the terminal; what follows is the actual content of the file `input.txt`).

```
% more input.txt
0 0 0 0 1 1 1 1
```

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

```
% java-introcs Mystery < input.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
```


Q6. 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. <code>false</code> | G. <code>x && y</code> | K. <code>x y</code> | O. <code>count++</code> |
| B. <code>true</code> | H. <code>x && z</code> | L. <code>x z</code> | P. <code>count--</code> |
| C. <code>x</code> | I. <code>y && z</code> | M. <code>y z</code> | Q. <code>count</code> |
| D. <code>y</code> | J. <code>x && y && z</code> | N. <code>x y z</code> | R. <code>count <= 1</code> |
| E. <code>z</code> | | | S. <code>count >= 2</code> |

A.

```
public static boolean majority(boolean x, boolean y, boolean z) {  
  
    int count = 0;  
  
    if (____) ____;  
  
    if (____) ____;  
  
    if (____) ____;  
  
    return ____;  
}
```

B.

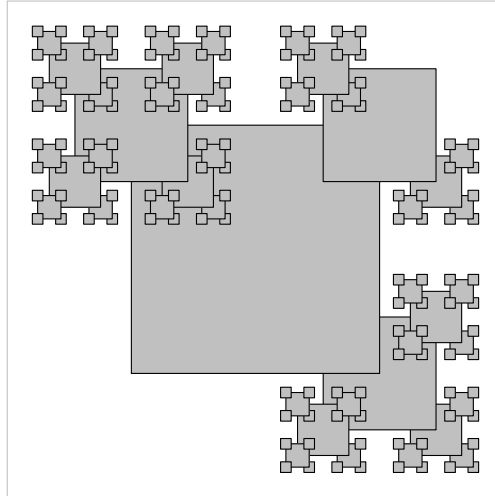
```
public static boolean majority(boolean x, boolean y, boolean z) {  
  
    if (____) return ____;  
  
    else if (____) return ____;  
  
    else if (____) return ____;  
  
    else return ____;  
}
```

Q7. Recursive Graphics (10 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 214th 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.

- Statement 1 appears first.
- Statement 2 appears after statement 5.
- Statement 2 appears before statements 3, 4, 5, and 6.
- Statements 3 and 6 appear before statements 4 and 5.
- If statements 3 and 6 are swapped, the function will still produce the same intermediate result.