

CSE 142, Spring 2009
Midterm Exam, Friday, May 8, 2009

Name: _____

Section: _____ **TA:** _____

Student ID #: _____

- You have 50 minutes to complete this exam.
You may receive a deduction if you keep working after the instructor calls for papers.
- This exam is open-book/notes. You may not use any computing devices including calculators.
- Code will be graded on proper behavior/output and not on style, unless otherwise indicated.
- Do not abbreviate code, such as “ditto” marks or dot-dot-dot ... marks.
- You do not need to write `import` statements in your code.
- If you enter the room, you must turn in an exam before leaving the room.
- You must show your Student ID to a TA or instructor for your exam to be accepted.

Good luck!

Score summary: (for grader only)

Problem	Description	Earned	Max
1	Expressions		10
2	Parameter Mystery		12
3	If/Else Simulation		12
4	While Loop Simulation		12
5	Assertions		15
6	Programming		15
7	Programming		15
8	Programming		9
X	Extra Credit		+1
TOTAL	Total Points		100

1. Expressions

For each expression in the left-hand column, indicate its value in the right-hand column.

Be sure to list a constant of appropriate type and capitalization.

e.g., 7 for an int, 7.0 for a double, "hello" for a String, true or false for a boolean.

<u>Expression</u>	<u>Value</u>
$3 * (7 + 1) - 3 - 7 + 2$	_____
$9 \% 5 + 20 \% 5 + 20 \% 3$	_____
$2.3 * 2.0 + 1.5 * 10 / 3$	_____
$1 + 2 + "3 + 4" + 5 + 6 / 4$	_____
$5 \% 2 + 10 / 5 / 2 + 4.2$	_____

2. Parameter Mystery

What output is produced by the following program?

```
public class ParameterMystery {
    public static void main(String[] args) {
        String head = "shoulders";
        String knees = "toes";
        String elbow = "head";
        String eye = "eyes and ears";
        String ear = "eye";

        touch(ear, elbow);
        touch(elbow, ear);
        touch(head, "elbow");
        touch(eye, eye);
        touch(knees, "Toes");
        touch(head, "knees " + knees);
    }

    public static void touch(String elbow, String ear) {
        System.out.println("touch your " + elbow + " to your " + ear);
    }
}
```

3. If/Else Simulation

For each call below to the following method, write the output that is produced, as it would appear on the console:

```
public static void ifElseMystery(int x, int y) {
    int z = 4;
    if (z <= x) {
        z = x + 1;
    } else {
        z = z + 9;
    }
    if (z <= y) {
        y++;
    }

    System.out.println(z + " " + y);
}
```

Method Call

Output

ifElseMystery(3, 20);

ifElseMystery(4, 5);

ifElseMystery(5, 5);

ifElseMystery(6, 10);

4. While Loop Simulation

For each call below to the following method, write the output that is produced, as it would appear on the console:

```
public static void whileMystery(int n) {
    int x = 0;
    int y = 1;
    while(n > x && n > y) {
        n--;
        x = x + 2;
        y = y + x;
    }
    System.out.println(x);
}
```

Method Call

whileMystery(5);

whileMystery(10);

whileMystery(4);

Output

5. Assertions

For each of the five points labeled by comments, identify each of the assertions in the table below as either being *always* true, *never* true, or *sometimes* true / sometimes false.

```
public static int assertionMystery(Scanner console) {
    int a = -1;
    int b = 0;
    int next = console.nextInt();
    // Point A
    while(next != 0) {
        // Point B
        b = Math.max(a, next);
        a = next;
        // Point C
        if(a == 42) {
            // Point D
            a++;
        }
        next = console.nextInt();
    }
    // Point E
    return b;
}
```

Fill in each box below with one of ALWAYS, NEVER or SOMETIMES. (You may abbreviate them as A, N, or S.)

	next == 0	a > 0	b >= next
Point A			
Point B			
Point C			
Point D			
Point E			

6. Programming

Write a program that simulates an ant trying to crawl up a building of height 6 steps. The ant starts on the ground, at height 0. Each iteration, the ant either crawls up one step, or slips off and falls all the way back to the ground. There is a 50% chance on each iteration that the ant will slip. The program should keep going until the ant gets to the top of the building. It should then print out the number of falls that the ant took before it finally reached the top.

Here is a sample execution:

```
number of falls: 8
```

7. Programming

Write a static method `isPowerOfTwo` that takes an integer `n` as an argument, and that returns `true` if `n` is a power of two, and otherwise `false`. If `n` is zero or negative, return `false`. Note that `isPowerOfTwo(1)` should return `true`, since $2^0=1$.

Here are some example calls to the method and the value it returns:

Call	<code>isPowerOfTwo(0)</code>	<code>isPowerOfTwo(1)</code>	<code>isPowerOfTwo(5)</code>	<code>isPowerOfTwo(8)</code>
Result	<code>false</code>	<code>true</code>	<code>false</code>	<code>true</code>

8. Programming

A “perfect number” is a positive integer that is the sum of all its proper factors (that is, factors including 1 but not the number itself). The first two perfect numbers are 6 and 28, since $1+2+3=6$ and $1+2+4+7+14=28$. Write a static method `findAllPerfectUpto` that takes an integer `max` as an argument and prints out all perfect numbers that are less than or equal to `max`.

Here is the console output from a call to `findAllPerfectUpto(6)`:

```
Perfect numbers up to 6: 6
```

Here is the console output from a call to `findAllPerfectUpto(500)`:

```
Perfect numbers up to 500: 6 28 496
```


X. Extra Credit (+1 point)

Make up an English word for a monster that is part Alan, part Dan. Use only the letters A, L, D, and N. Example: Landalad. Do not use this example.

(Any other word you write will get the +1 extra point.)