

CSE 142, Autumn 2007

Midterm Exam Key

1. Expressions

<u>Expression</u>	<u>Value</u>
$3 + 4 * 5 / 2$	13
$13 \% 5 + 43 \% (11 \% 3)$	4
$1.5 * 3.0 + 25.0 / 10.0$	7.0
$5 / 2 + 123 / 10 / 10.0$	3.2
$5 + 2 + "(1 + 1)" + 4 + 2 * 3$	"7(1 + 1)46"

2. Parameter Mystery

troops gave cause to the support
cause gave support to the troops
rudy gave hillary to the cause
hillary gave troops to the p
support gave support to the cause

3. While Loop Simulation

<u>Method Call</u>	<u>Value Returned</u>
mystery(2);	1
mystery(-1);	0
mystery(7);	3
mystery(18);	2
mystery(43);	4

4. Assertions

	<u>flip == 0</u>	<u>heads == 0</u>	<u>flip > heads</u>
Point A	NEVER	ALWAYS	ALWAYS
Point B	SOMETIMES	SOMETIMES	SOMETIMES
Point C	ALWAYS	NEVER	NEVER
Point D	NEVER	SOMETIMES	SOMETIMES
Point E	ALWAYS	NEVER	NEVER

5. Programming (four solutions shown)

```
public static boolean monthApart(int m1, int d1, int m2, int d2) {
    if (m1 == m2) {
        return false;
    } else if (m1 <= m2 - 2) {
        return true;
    } else if (m1 >= m2 + 2) {
        return true;
    } else if (m1 == m2 - 1) {
        if (d1 <= d2) {
            return true;
        } else {
            return false;
        }
    } else if (m1 == m2 + 1) {
        if (d1 >= d2) {
            return true;
        } else {
            return false;
        }
    } else {
        return false;
    }
}

public static boolean monthApart(int m1, int d1, int m2, int d2) {
    if (m1 < m2 - 1 || m1 > m2 + 1) {
        return true;
    } else if (m1 == m2 - 1 && d1 <= d2) {
        return true;
    } else if (m1 == m2 + 1 && d1 >= d2) {
        return true;
    } else {
        return false;
    }
}

public static boolean monthApart(int m1, int d1, int m2, int d2) {
    return (m2 - m1 > 1) || (m1 - m2 > 1) ||
           (m2 - m1 == 1 && d1 <= d2) ||
           (m1 - m2 == 1 && d1 >= d2);
}

public static boolean monthApart(int m1, int d1, int m2, int d2) {
    return Math.abs((m1 * 31 + d1) - (m2 * 31 + d2)) >= 31;
}
```

6. Programming (one solution shown)

```
public static void sequenceSum(double limit) {
    if (limit >= 1) {
        System.out.print("1");
        int denominator = 1;
        double sum = 1.0;
        while (sum < limit) {
            denominator++;
            sum += 1.0 / denominator;
            System.out.print(" + 1/" + denominator);
        }
        System.out.println(" = " + sum);
    }
}
```