CSE 142, Winter 2007
Midterm Exam Key

1. Expressions (10 points)

Expression	Value
`1 + 2 * 3 - 4 * 5`	`-13`
`5 / 2 + 9.0 / 2.0 - 2 * 1.25`	`4.0`
`29 % 2 % 5 + 34 % 3`	`2`
`8 + 6 * -2 + 4 + "0" + (2 + 5)`	`"007"`
`31 / 2 / 10.0 + 10 / (5 / 2.0)`	`5.5`

2. Parameter Mystery (20 points)

```
drew saw the felt
sue felt the saw
sue drew the b
b sue the a
drew felt the felt
```

3. While Loop Simulation (15 points)

Method Call	Output
`mystery(42, 0);`	`42`
`mystery(6, 12);`	`12 6 6`
`mystery(18, 27);`	`27 18 9 9`
`mystery(24, 60);`	`60 36 24 12 12`
`mystery(50, 15);`	`50 35 20 15 10 5 5`

4. Assertions (15 points)

	`x > 2`	`x < n`	`n % x == 0`
Point A	NEVER	SOMETIMES	SOMETIMES
Point B	SOMETIMES	ALWAYS	SOMETIMES
Point C	NEVER	SOMETIMES	SOMETIMES
Point D	ALWAYS	SOMETIMES	SOMETIMES
Point E	SOMETIMES	NEVER	SOMETIMES

5. Programming (15 points) -- five solutions shown

```
public static boolean enoughTimeForLunch(
        int h1, int m1, int h2, int m2) {
    if (h1 > h2) {
        return false;
    } else if (h1 == h2) {
        return m2 - m1 >= 45;
    } else if (h1 == h2 - 1) {
        return 60 + m2 - m1 >= 45;
    } else {
        return true;
    }
}
```

```
public static boolean enoughTimeForLunch(
        int h1, int m1, int h2, int m2) {
    if (h2 > h1 + 1) {
        return true;
    } else if (h2 == h1 && m1 + 45 <= m2) {
        return true;
    } else if (h2 == h1 + 1 && m1 - 15 <= m2) {
        return true;
    } else {
        return false;
    }
}
```

```
public static boolean enoughTimeForLunch(int h1, int m1, int h2, int m2) {
    if (h1 > h2) {
        return false;
    } else if (h1 == h2) {          // same hour
        if (m1 + 45 <= m2) {        //   must be >= 45 min apart
            return true;
        } else {
            return false;
        }
    } else if (h2 == h1 + 1) {      // h1 is just before h2
        if (m1 - 15 <= m2) {        //   must be >= -15 min apart
            return true;
        } else {
            return false;
        }
    } else {                        // time 1 is > 1 hour before time 2
        return true;
    }
}

public static boolean enoughTimeForLunch(int h1, int m1, int h2, int m2) {
    if ((h1 == h2 && m1 + 45 <= m2) || (h2 == h1 + 1 && m1 - 15 <= m2) || (h1 < h2 - 1)) {
        return true;
    } else {
        return false;
    }
}

public static boolean enoughTimeForLunch2(int h1, int m1, int h2, int m2) {
    return 60 * h1 + m1 + 45 <= 60 * h2 + m2;
}
```

6. Programming (15 points) -- three solutions shown

```
public static void printGrid(int rows, int cols) {
    for (int i = 1; i <= rows; i++) {
        System.out.print(i);
        for (int j = 1; j <= cols - 1; j++) {
            System.out.print(", " + (i + rows * j));
        }
        System.out.println();
    }
}
```

```
public static void printGrid(int rows, int cols) {
    for (int i = 1; i <= rows; i++) {
        for (int j = 0; j < cols - 1; j++) {
            System.out.print((i + rows * j) + ", ");
        }
        System.out.println(i + rows * (cols - 1));
    }
}
```

```
public static void printGrid(int rows, int cols) {
    int n = 1;
    int count1 = 1;
    int count2 = 1;
    while (count1 <= rows * cols) {
        if (count1 % cols == 0) {
            System.out.println(n);
            count2++;
            n = count2;
        } else {
            System.out.print(n + ", ");
            n = n + rows;
        }
        count1++;
    }
}
```

7. Programming (10 points) -- three solutions shown

```
public static void favoriteLetter(Scanner console, String letter) {
    System.out.println("Looking for two \"" + letter + "\" words in a row.");
    int count = 0;
    System.out.print("Type a word: ");
    String word = console.next();
    while (count < 2) {
        if (word.startsWith(letter)) {
            count++;
        } else {
            count = 0;
        }
        System.out.print("Type a word: ");
        word = console.next();
    }
    System.out.println("\"" + letter + "\" is for \"" + word + "\"");
}

// uses two Strings instead of count, and uses forever/break loop
public static void favoriteLetter(Scanner console, String letter) {
    System.out.println("Looking for two \"" + letter + "\" words in a row.");
    System.out.print("Type a word: ");
    String word1 = console.next();
    System.out.print("Type a word: ");
    String word2 = console.next();
    while (!(word1.startsWith(letter) && word2.startsWith(letter))) {
        word1 = word2;
        System.out.print("Type a word: ");
        word2 = console.next();
    }
    System.out.println("\"" + letter + "\" is for \"" + word2 + "\"");
}

// uses do/while loop
public static void favoriteLetter(Scanner console, String letter) {
    System.out.println("Looking for two \"" + letter + "\" words in a row.");
    int count = 0;
    String word;
    do {
        System.out.print("Type a word: ");
        word = console.next();
        if (word.startsWith(letter)) {
            count++;
        } else {
            count = 0;
        }
    } while (count < 2);
    System.out.println("\"" + letter + "\" is for \"" + word + "\"");
}
```