

CSE142 Midterm Key  
Autumn 2019

1.	Expression	Value
	$43 \% 15 / 3 + 15 / 2$	11
	$6.2 * 5 / 10 + 3.5$	6.6
	$6 * 2.5 / 4 + (2.3 + 2.7) / 4$	5.0
	$"18" + 3 * 4 + (8 + 5)$	"181213"
	$59 \% 10 / (2 + 2) * 2.5 / 2$	2.5

2. The program produces the following output:

```

one fish, two fish
blue red, fish fish
red one, two fish
red fish, blue fish

```

3.	Method Call	Output Produced
	<code>ifElseMystery(3, 1);</code>	13 1 2
	<code>ifElseMystery(6, 9);</code>	16 3 5
	<code>ifElseMystery(5, -1);</code>	5 9 5
	<code>ifElseMystery(1, 2);</code>	1 12 4

4.	Method Call	Output Produced
	<code>mystery(5, 7);</code>	2 -2 -2
	<code>mystery(4, 20);</code>	16 13 11 10 10
	<code>mystery(10, 40);</code>	30 21 13 6 0 0
	<code>mystery(5, 15);</code>	10 6 3 1 0 0

5.	a != 0	c % 2 == 0	b > 0
Point A	sometimes	always	never
Point B	always	sometimes	sometimes
Point C	always	never	never
Point D	sometimes	sometimes	sometimes
Point E	never	sometimes	sometimes

6. One possible solution is shown below.

```
public static int noBigger(int max, Random rand) {
    System.out.println("Picking numbers from 1 to " + max);

    int num = rand.nextInt(max) + 1;
    System.out.println("Number: " + num);
    int last = num;
    int count = 1;
    while (num <= last) {
        double probab = (double)num / max;
        System.out.println("Probability to continue: " + probab);
        last = num;
        num = rand.nextInt(max) + 1;
        System.out.println("Number: " + num);
        count++;
    }
    System.out.println("Streak ends");
    return count;
}
```

7. One possible solution is shown below.

```
public static void invest(Scanner console, double init, int years) {
    System.out.println("Starting with: $" + init);
    double curr = init;
    for (int i = 1; i <= years; i++) {
        System.out.print("This year's return? ");
        int rate = console.nextInt();
        curr *= (1 + rate / 100.0);
        System.out.println("After year " + i + ": $" + curr);
    }
    System.out.println("Total interest earned: $" + (curr - init));
}
```

8. One possible solution is shown below.

```
public static int longestStreak(String str) {
    int max = 1;
    int len = 1;
    char curr = str.charAt(0);

    for (int i = 1; i < str.length(); i++) {
        if (str.charAt(i) == curr) {
            len++;
            max = Math.max(max, len);
        } else {
            len = 1;
            curr = str.charAt(i);
        }
    }

    return max;
}
```