

CSE 142, Spring 2009, Sample Final Exam 2, Sample Solutions

1. Array Simulation

```
{}  
{0}  
{0, 2}  
{0, 4, 6}  
{0, 4, 12, 24}
```

2. Inheritance

```
b  
c 1  
a 2
```

```
b  
c 1  
b 2
```

```
c  
c 1  
c 2
```

```
b  
d 1  
b 2
```

3. Parameters and References

```
2 [0, 0, 1, 0]  
1 [0, 0, 1, 0]  
3 [0, 0, 1, 1]  
2 [0, 0, 1, 1]
```

4. Token-Based File Processing. One possible solution appears below.

```
public static double evaluate(Scanner input) {  
    double result = input.nextDouble();  
    while (input.hasNext()) {  
        String operator = input.next();  
        double num = input.nextDouble();  
        if (operator.equals("+")) {  
            result += num;  
        } else { // operator.equals("-")  
            result -= num;  
        }  
    }  
    return result;  
}
```

```
}
```

5. Line-Based File Processing. One possible solution appears below.

```
public static void processScores(Scanner input) {
    while (input.hasNextLine()) {
        String name = input.nextLine();
        String data = input.nextLine();
        int plus = 0;
        int count = 0;
        for (int i = 0; i < data.length(); i++) {
            count++;
            if (data.charAt(i) == '+') {
                plus++;
            }
        }
        double percent = 100.0 * plus / count;
        System.out.println(name + ": " + percent + "% plus");
    }
}
```

6. Arrays. One possible solution appears below.

```
public static boolean isFibLike(int[] list) {
    for (int i = 2; i < list.length; i++) {
        if (list[i] != list[i - 2] + list[i - 1]) {
            return false;
        }
    }
    return true;
}
```

7. ArrayList. One possible solution appears below.

```
public static void removeShorterStrings(ArrayList<String> list) {
    for (int i = 0; i < list.size() - 1; i++) {
        String first = list.get(i);
        String second = list.get(i + 1);
        if (first.length() <= second.length()) {
            list.remove(i);
        } else {
            list.remove(i + 1);
        }
    }
}
```

8. Critters. One possible solution appears below.

```
public class Pigeon extends Critter {
    private String display;
    Random r;

    public Pigeon() {
        display = "*";
    }
}
```

```

        r = new Random();
    }

    public Action getMove(CritterInfo info) {
        if (info.getFront() == Neighbor.EMPTY) {
            display = "H";
            return Action.HOP;
        } else {
            int flip = r.nextInt(2);
            if (flip == 0) {
                display = "L";
                return Action.LEFT;
            } else {
                display = "R";
                return Action.RIGHT;
            }
        }
    }

    public String toString() {
        return display;
    }
}

```

9. Arrays. One possible solution appears below.

```

public static int[] append(int[] list1, int[] list2) {
    int[] result = new int[list1.length + list2.length];
    for (int i = 0; i < list1.length; i++) {
        result[i] = list1[i];
    }
    for (int j = 0; j < list2.length; j++) {
        result[list1.length + j] = list2[j];
    }
    return result;
}

```

10. Programming. One possible solution appears below.

```

public static boolean isUnique(int[] list) {
    for (int i = 0; i < list.length; i++) {
        for (int j = i + 1; j < list.length; j++) {
            if (list[i] == list[j]) {
                return false;
            }
        }
    }
    return true;
}

```