

## Solutions to Practice Final #2

Based on a past version of this handout by Marty Stepp and Keith Schwarz

### 1. Value/Reference Semantics Trace (15 points)

```
2 3 [0, 0, 17, 0]
3 1 [0, 0, 17, 0]
1 0 [17, 0, 17, 0]
0 1 [17, 0, 17, 0]
```

### 2. Multi-dimensional Arrays Trace (15 points)

a)

```
[[4, 5, 6, 6],
 [4, 6, 7, 1],
 [1, 6, 7, 2]]
```

b)

```
[[2, 3, 6, 5, 5],
 [4, 6, 8, 10, 10],
 [9, 5, 1, 2, 4]]
```

### 3. Collections Trace (15 points)

```
{deux=two, cinq=five, one=un, three=trois, four=quatre}
{board=skate, car=drive, computer=program, awesome=computer}
{ebert=siskel, boy=girl, heads=tails, begin=ready, first=last}
```

### 4. ConsoleProgram (30 points)

```
public int neverEndingBirthdayParty() {
    RandomGenerator rgen = RandomGenerator.getInstance();
    boolean[] used = new boolean[366];
    int numLeft = 366;
    int numPeople = 0;
    while (numLeft > 0) {
        int birthday = rgen.nextInt(0, 365);
        if (!used[birthday]) {
            numLeft--;
            used[birthday] = true;
        }
        numPeople++;
    }
    return numPeople;
}
```

## 5. Arrays (20 points)

```
public int longestSortedSequence(int[] list) {
    if (list.length == 0) {
        return 0;
    }

    int max = 1;
    int count = 1;
    for (int i = 1; i < list.length; i++) {
        if (list[i] >= list[i - 1]) {
            count++;
        } else {
            count = 1;
        }

        if (count > max) {
            max = count;
        }
    }
    return max;
}
```

## 6. Classes and Objects (25 points)

```
// solution 1
public boolean transactionFee(double amount) {
    for (int i = 1; i <= transactions; i++) {
        balance -= amount * i;
    }

    if (balance > 0.0) {
        return true;
    } else {
        balance = 0.0;
        return false;
    }
}

// solution 2
public boolean transactionFee(double amount) {
    for (int i = 1; i <= transactions; i++) {
        balance = Math.max(0.0, balance - amount * i);
    }
    return balance > 0.0;
}
```

## 7. Collections (30 points)

```
public boolean isSubMap(HashMap<String, String> map1,
                       HashMap<String, String> map2) {
    for (String key : map1.keySet()) {
        if (!map2.containsKey(key) || !map1.get(key).equals(map2.get(key))) {
            return false;
        }
    }
    return true;
}
```

## 8. Graphical User Interfaces (30 points)

```
public class SignMaker extends GraphicsProgram {
    private int labelY;
    private JTextField line;
    private JTextField font;

    public void init() {
        line = new JTextField(30);
        line.addActionListener(this);
        font = new JTextField(15);
        font.setText("Times-Bold-36");
        labelY = 0;
        add(new JLabel("Line: "), SOUTH);
        add(line, SOUTH);
        add(new JLabel(" Font: "), SOUTH);
        add(font, SOUTH);
    }

    public void actionPerformed(ActionEvent e) {
        if (e.getSource() == line) {
            GLabel label = new GLabel(line.getText());
            label.setFont(font.getText());
            labelY += label.getHeight();
            double x = (getWidth() - label.getWidth()) / 2;
            add(label, x, labelY);
            line.setText("");
        }
    }
}
```