

SURNAME: FORENAMES:

CompSci 101 AC 2002 Final Exam Model Answers**QUESTION 1**

- a) -1
- b) 15.105
- c) d is: 12.0
- d) i is : 123
- e) 2

QUESTION 2

```
private static String randomString(String[] words) {
    int randIndex = (int)(Math.random() * words.length);
    return words[randIndex];
}
```

QUESTION 3

```
private static int[][] flipHorizontal(int[][] a){
    int arrayRows = a.length;
    int arrayCols = a[0].length;
    int[][] b = new int[arrayRows][arrayCols];
    for(int i=0;i<arrayRows; i++){
        for(int j=0; j<arrayCols ; j++){
            b[i][j] = a[arrayRows-i-1][j];
        }
    }
    return b;
}
```

QUESTION 4

```
public void actionPerformed(ActionEvent e) {
    String text = tArea.getText();
    String word = tInput.getText();
    int index = text.indexOf(word);
    if (index > -1)
        tArea.setText(text.substring(0, index+word.length()));
}
```

SURNAME: FORENAMES:

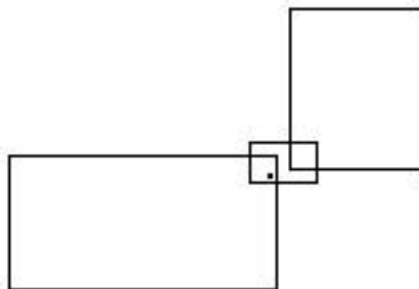
QUESTION 5

```
public void mousePressed(MouseEvent e) {
    for (int i = 0; i < NUM_RECTANGLES; i++)
        if (boxes[i].contains(e.getPoint()))
            crosses[i] = !crosses[i];
    repaint();
}

public void paint(Graphics g) {
    for (int i = 0; i < NUM_RECTANGLES; i++) {
        g.drawRect(boxes[i].x, boxes[i].y, boxes[i].width,
                  boxes[i].height);
        if (crosses[i]) {
            g.drawLine(boxes[i].x, boxes[i].y, boxes[i].x +
                      boxes[i].width, boxes[i].y + boxes[i].height);
            g.drawLine(boxes[i].x + boxes[i].width,
                      boxes[i].y, boxes[i].x, boxes[i].y +
                      boxes[i].height);
        }
    }
}
```

QUESTION 6A
C
E
F

Here is a diagram of the point and the rectangles:

**QUESTION 7**

```
> java Q7
AA
i: 4, j: 0, value: 0
i: 4, j: 3, value: 1
AA
value: 3
```

SURNAME: FORENAMES:

QUESTION 8

There are lots of different ways of doing this. Below is one possible way:

```
public class Q8 extends Applet {
    public void paint(Graphics g) {
        final int LINES = 7;
        final int SIZE = 20;
        int toDoAcross = LINES;
        int startX = 20;
        int y = 20;
        int x;

        for (int row = 0; row < LINES; row++) {
            x = startX + SIZE * row;
            y = x;
            for (int col = 0; col < toDoAcross; col++) {
                if (row % 2 == 0)
                    g.fillOval(x, y, SIZE, SIZE);
                else
                    g.fillRect(x, y, SIZE, SIZE);
                x += 2*SIZE;
            }
            toDoAcross--;
        }
    }
}
```

QUESTION 9

```
public class Q9 {
    /* This program prints a greeting */
    public static void main(String[] args) {
        System.out.println("Hello world!");
        if (Math.random < 0.5)
            System.out.println("Goodbye");
    }
}
```

Error 1: The ending comment symbol should be */

Error 2: Math.random should be Math.random()

QUESTION 10

13: word: GO SOFTLY value: 9

13: word: STOP value: 4

SURNAME: FORENAMES:

QUESTION 11

```
public class DemoCall{
    private int hour;
    private int minute;
    private int responseMin;
    private int qNum;
    private String login;
    private boolean isProcessed;
    //-----
    //-----
    public DemoCall(int hr, int min, String log, int qN){
        hour = hr;
        minute = min;
        login = log;
        qNum = qN;
        isProcessed = false;
    }
    //-----
    //-----
    public void callProcessed(int hr, int min){
        responseMin = (hr-hour)*60;
        responseMin += min - minute;
        isProcessed = true;
    }
    //-----
    //-----
    public void moveUpQ(){
        if(qNum > 0)
            qNum--;
    }
    //-----
    //-----
    public boolean getIsProcessed(){
        return isProcessed;
    }
    //-----
    //-----
    public boolean isBeforeInQ(DemoCall otherD){
        if (qNum > otherD.qNum)
            return true;
        return false;
    }
    //-----
    //-----
    public String toString(){
        String temp;

        temp = login + " " + hour+"."+minute;
        if (isProcessed)
            temp += " RESPONSE "+responseMin+" MINUTES";
        else
            temp += " NOT PROCESSED: "+qNum + " IN QUEUE";
        return temp;
    }
}
```