

Name: _____

**CS110 Introduction to Computing
Spring 2011
Exam 2**

This is a closed book / closed notes exam. Your work on this exam must be your own. Answer all questions in the space provided, continuing your answer on the back of the page if necessary. If you want scrap paper, please ask your instructor.

All Processing statements you write should be syntactically correct. There will be no credit for incorrect use of Processing syntax.

Be certain to use proper indentation and commenting in any code you write.

Be certain to sign the log-in sheet for the examination period, and the statement of academic honesty on the exam cover page.

I wish you each the best of luck!

Sign the following statement **after** you have completed the examination. Your exam will **not be graded** without your signature.

I certify that my responses in this examination are solely the product of my own work and that I have fully abided by the Bryn Mawr College Computer Science Academic integrity policy and instructions stated above while taking this exam.

Signature: _____

Printed Name: _____

For Instructor Use Only:

Question #	Points	Maximum Points
1.		10
2.		15
3.		10
4.		10
5.		15
6.		10
7.		15
8.		20
Total		100

Name: _____

Section 1: String Manipulation (10 points total)

Circle or write the correct answer to each question.

1.1 (1 pt) Write your name on EVERY page of the exam booklet and any scrap paper you use.

Given the commands:

```
String aPalindrome = "a man, a plan, a canal Panama";  
String[] strs = splitTokens(aPalindrome, ",");
```

Answer the following questions:

1.2 (3 pts) What will be the length of `strs`?

- a) 1
- b) 2
- c) 3
- d) 4

1.3 (3 pts) What will be the value of `strs[1]`?

- a) "a man"
- b) "a plan"
- c) "a canal Panama"
- d) 3

1.4. (3 pts) Write the expression used to obtain the number of elements in `strs`.

Section 2: Conditionals (15 points total)

2.1 (7 pts) Write a conditional expression that would print to the screen "weekday" whenever the integer variable `day` is not equal to 0 or 6.

2.2 (8 pts) Consider a function which toggles (i.e., switches) the value for a global boolean variable that tracks whether a light is on or off. Two people, creatively named A and B, who are claiming to be expert light switchers have written two different implementations of this function, shown below:

```
// the global variable
boolean lightOn = false;
...
void toggleLightA() {
    if (lightOn) {
        lightOn = false;
    } else {
        lightOn = true;
    }
}
```

```
// the global variable
boolean lightOn = false;
...
void toggleLightB() {
    if (lightOn) {
        lightOn = false;
    }
    if (!lightOn){
        lightOn = true;
    }
}
```

Are these two implementations equivalent? Are both of the people expert light switchers, or is one or are both of them frauds? Why? Explain your answer!

Name: _____

Section 3: Multidimensional Arrays (10 points total)

3. Add the necessary lines of code within `setup()` that fills the `vals` array with random numbers of your choosing. Your implementation must use `for` loops.

```
float[][] vals;  
  
void setup() {  
    vals = new float[20][300];  
  
    // Add your code here
```

```
} // Closing brace for setup()
```

Name: _____

Section 4: Transformations (10 points total)

4. Add the necessary transformations to `draw()` to render the rectangle at the center of the sketch, twice its size, and rotated by 45 degrees ($\pi/4$ radians).

```
void setup() {
    size(400, 400);
    rectMode(CENTER);
}

void draw() {

    // Add transformations here

    rect(0, 0, 25, 25);
}
```

Name: _____

Section 5: Sorting and Searching (15 points total)

Consider the following array:

```
float[] vals = new float[]{ 1, 3, 6, 8, 9, 13, 19, 23, 32, 40 };
```

We could use the following code to determine whether the value x is in the array:

```
float x = 10;
boolean containsValue = false;
for (int i=0; i < vals.length; i++) {
    if (vals[i] == x) { // comparison
        containsValue = true;
    }
}
```

However, in the worst case, this method requires `vals.length` (10) comparisons.

5.1 (10 pts) Describe *in detail* how the binary search algorithm would work to find whether x is in the array. Make certain to describe how the algorithm works.

5.2 (5 pts) How many comparisons would binary search take to solve the same problem? Justify your answer if you're not certain.

Section 6: Image Processing (10 points total)

6. Add the necessary code within the nested for-loops to color all pixels white, except for pixels on the diagonal (when $r == c$).

```
void setup() {  
    size(100,100);  
  
    loadPixels();  
  
    for (int r=0; r<width; r++) {  
        for (int c=0; c<height; c++) {  
  
            // Add code here  
  
        } // Closing brace for the c-loop  
    } // Closing brace for the r-loop  
  
    updatePixels();  
}
```

Name: _____

Section 7: Recursion (15 points total)

7.1. (10 pts) What does the following program print? Justify your answer if you're not certain.

```
void setup() {  
    int n = op(5,3);  
    println( n );  
}  
  
int op (int val, int divisor) {  
    if (val < divisor) {  
        return val;  
    } else {  
        return op(val - divisor, divisor);  
    }  
}
```

7.2. (5 pts) Which built-in mathematical operator does the `op()` function simulate?

Section 8: Algorithms and Debugging (20 points total)

8. The following program was designed to count and print the number of duplicates in the myArray String array. Unfortunately, it doesn't work properly. When I test it with the given data, it tells me that I have 11 duplicates, but I know that there are only two. Fix the program so that it works correctly.

```
// Count and print the number of duplicate strings in myArray

String [] myArray = {"A", "B", "C", "D", "A", "F", "C"};

void setup() {

    int count = 0;

    for (int i=0; i<myArray.length; i++) {

        for (int j=0; j<myArray.length; j++) {

            if (myArray[i].equals(myArray[j])) {

                count++;

            }

        }

    }

    println("There are " + count + " duplicates.");
}
```