ISE 108 Practice Final Exam Instructions

1. Please sit at the machine to which you have been assigned. You can find the machine number on a label on the lid of each computer.

2. Please print your name NEATLY on the attendance sheet, next to the number of your assigned computer.

3. The score for a perfect final exam, not counting extra credit, is 150 points. Choose any combination of problems from your assigned problem set to add up to this point value.

4. Partial credit may be assigned at the instructor's discretion. Problems that do not compile will automatically receive 0 points, and WILL NOT be considered for partial credit.

5. The exam is open-book, open-notes, limited Internet. You will only have access to one Web site during the exam: the course Web page, which is located at http://www.cs.stonybrook.edu/~tashbook/fall2015/ise108

6. You may not use cell phones, tablets, e-readers, pagers, or any other electronic devices during the exam, except for USB flash drives and the computers that are installed in the exam room. Please turn all prohibited devices off for the duration of the exam. You may not communicate with any other students during the exam. Any suspected cheating will result in a grade of 0 for the exam and academic dishonesty charges.

7. At the start of the exam, your instructor will tell you where to find the questions and the starting code for the exam problems.

8. You will be provided with instructions as to how to submit your work for the exam. We will ONLY grade files that have been submitted via this method. If you are unsure how to do this, ask the instructor or one of the TAs for assistance.
1. Strings and Text Manipulation (30 points)

The "Vowels" program prompts the user to enter some text. Complete the countVowels() function, which counts and prints the number of a's, e's, i's, o's, and u's in that text. Your solution should print each vowel's frequency separately.

2. Arrays (30 points)

Complete the "Interleave" program, which takes two arrays (of equal length) and returns a single large array that contains the contents of the source arrays, in alternating order. For example, if the input arrays are {1, 3, 5, 7, 9} and {2, 4, 6, 8, 10}, the resulting interleaved array should be {1, 2, 3, 4, 5, 6, 7, 8, 9, 10}.

3. Classes (30 points)

In a new Processing file, define a class named Value, with the following:

- a boolean instance variable named modified
- an int instance variable named val
- a no-argument method named getVal() that returns the current value of val.
- a method named setVal() that does not return any value. This method takes an integer argument and assigns it to val, then sets modified to true.
- a boolean method wasModified() that returns the value of modified

4. Recursion (30 points)

The obvious way to compute $a^n$ for some integer $n$ is to multiply $a$ by itself $n$ times (ex. $a^n = a \times a \times \cdots \times a$). This is easy to implement using a loop, but it is somewhat inefficient.

A more efficient approach uses a recursive “divide and conquer” approach. Suppose that we want to compute $2^8$. This is equivalent to $2^4(2^4)$. If we know what $2^4$ is, we only need to perform one more multiplication to get $2^8$. Likewise, $2^4$ is just $2^2(2^2)$. This gives us the basic formula

$$a^n = a^{n/2}(a^{n/2})$$

This formula only works for even exponents. To accommodate odd exponent values as well, we simply multiply one more time by $a$:

$$a^n = a^{n/2}(a^{n/2}) \times a \text{ if } a \text{ is odd}$$

We can use this information to complete the "Power" program, which recursively computes the nth power of any non-zero integer. For your base case, use the fact that any non-zero value raised to the 0th power is 1.

In order for you to receive ANY credit for this problem, your solution MUST use recursion in a non-trivial way to compute the solution! DO NOT use any of the built-in Java or Processing commands to perform the exponentiation operation.
5. Sorting (30 points)

On the paper provided, show the round-by-round results of performing *selection sort* on the following list of values:

13, 28, 66, 86, 78, 33, 99, 55, 34, 8

6. Drawing (30 points)

Create a Processing sketch that draws a snowman. Your snowman should consist of at least three circles and have a face and arms. Your snowman should also have one accessory (like a pipe or a hat).