

## Extra Credit Project

This extra credit project will be worth three points added to your last test grade, and will be due at 11:59pm on Friday, May 6<sup>th</sup>.

For this project, you will need to learn how to use the Julia programming language. Your primary resource for this will be the website <http://julia.org> (short for Julia language). Start by going to the “Downloads” section and picking the right version of the current release for your operating system. Once you’ve downloaded it, opening the program should open up a command window with the Julia logo and version information on it.

Now that you’ve downloaded Julia, you’ll need to learn how to use it. Start by going to the “Docs” (short for “Documentation”) section on the Julia website. You can skip the “Introduction” section, as that is more a comparison between Julia and other programming languages, and will only make sense if you have experience with other programming languages. Start with the “Getting Started” section, and with the resources given at the bottom of that page. Then go on to “Variables” and keep going down the list of sections until at least the “Types” section. There will be a lot of stuff that doesn’t make sense to you on your first reading, so you might have to go back and read some of this material a second time after getting your bearings. While reading through this, try out the examples in your command window with Julia running to experiment with how things work.

For larger or more involved problems, you’ll want to use a text editor to hold your work. You can work in the text editor and then copy and paste lines of code into the command window with Julia running (This is called the REPL, for read-evaluate-print-loop.). This way you can save your work as well.

Once you have a very basic idea of how programming languages work, and how the Julia programming language works in particular, you’ll want to start with a warm-up problem:

**Problem Warm-up** Write a program in your text editor that, when the whole thing is copy and pasted into the REPL, counts how many numbers there are that are less than or equal to 100 and are multiples of 3 or 5 (Look up the “mod” function to deal with “multiples of”).

See if you can do this problem by yourself, but if you are having trouble here is my solution.

```
sum = 0
for i = 1:100000
    if mod(i, 3) == 0 || mod(i, 5) == 0
        sum = sum + i
    end
end
println(sum)
```

which, when run, prints the output 2418.

Now for your actual assignment: solve **ONE** of the following problems in Julia:

**Problem 1** Write a program that implements the nearest neighbor algorithm. You'll have to represent a complete, weighted graph in Julia as an array of vertices, an array of edges, and an array of weights on the edges. Your program should be able to accept a complete weighted graph of any number of vertices and with any positive weights.

**Problem 2** Write a program that implements the adjusted winner procedure. Your program should be able to accept any number of items being divided and any point distribution given. Your program should output which items are given to which party, including the fraction of the shared item given to each.

**Problem 3** Write a program that implements Jefferson's method. Your program should be able to accept a house size of any number, any number of states, and any population numbers, and then solve for the apportionment. You will have to use the bisection method or something similar to guess the correct divisor values, along with some method of choosing your first and second guesses for the divisor.

A note on grading: I will grade your assignment out of three points (the points to be added to your last test grade), depending on how satisfactory your program is. Note that I will pick some test problems to test your program, but you won't know what these test problems are before hand, so make sure your program works for multiple different examples. Also, make sure that any code you turn in is **YOUR OWN**. You can talk about the assignment with other students, and you may even work with other students, but I will be able to tell if you didn't write your own code from scratch. To turn in the assignment, you will email me a .txt file (or you can change the file extension to .jl, for a Julia program) with your program in it.