

## **Programming Homework 1**

### **Team Reflection Section**

Using MATLAB was easy in that the contents of the data file could be easily sorted into a matrix and then managed appropriately. However, the File I/O constructs were tricky to recall and execute. We learned about the `csvread` function in MATLAB to most easily sort data into a matrix from a file. Overall, the ease of managing matrices within MATLAB was best suited to this type of problem.

In python, it is impossible to read in numbers out of a formatted list, so one must read in the lines as strings and separate the numbers out by checking the value of the character at every location of the string. One positive aspect of python was that, one does not have to initialize every counting variable that is used.

Using C, it was fairly easy to implement the programming algorithm. One advantage of C is that it handles input very well. It has a variety of functions to accept numerical input and work with files. C also deals with error handling very well. If an error was encountered, the program can return a -1 and the program will terminate with a useful message. However, C is not as good as MATLAB at handling data and is rather particular with its syntax.

### **Individual Section**

While programming my solution to this problem, I discovered how to use the `csvread` function to sort the read-in file into a matrix. This made the input of the file data much easier and faster than my original idea, which was to read the file line by line as the data was calculated for each seat. Once the data was sorted into the matrix, the data calculation was easily done, as each entry in the matrix was able to be accessed much more easily than if the entries were accessed directly from the file. The most difficult part of the solution was working with the file I/O logic. In every

language, that is the construct in which I am least confident. Once the file data was input, the calculations were easily performed and the data output. In solving other problems similar to this, I would recommend using the `csvread` and other similar functions in order to make the input process of data in comma separated files easier. This also allows for an easier experience in handling the data and performing the calculations, as the file's contents are automatically sorted into a properly-sized matrix, which is easily managed within MatLab.

### Response to Question Section

#### High Confidence Programming Constructs

- Math functions
- Basic input / output
- Arrays / 2D Arrays
- Logical Variables
- If-else
- Relational Operators
- Logical Operators
- While Loops

#### Low Confidence Programming Constructs

- File I/O
- Functions