

# CSC 415 Fall 2020

## Assignment 1 – Application Development

Due: Monday September 14, 2020 by 11:59 p.m.

Grade: 100 points as per the rubric; 10 points penalty for each day late.

You must complete, i.e. design and develop the solution for, this assignment **individually**. You are encouraged to ask general questions about the requirements, design and programming concepts, or compile error messages in class. You are encouraged to discuss questions or concerns specific to your solution with Dr. Pulimood during office hours, on Discord, or via email. You can also ask Heavenly for help on using git, GitHub, the VM, and Ruby. You may work with each other to learn the Ruby language. Seeking assistance from, or providing assistance to, other students on solving the problem and writing the program is a violation of TCNJ's Academic Integrity policy.

### Objectives:

The main goals of this assignment are for you to demonstrate that you have a good understanding of prerequisite computer science concepts, and the ability to design and implement a computational solution for a problem based on the requirements provided. Specifically, you should demonstrate:

- An ability to solve a problem, based on specifications and requirements provided,
- A good grasp of fundamental programming concepts like loops, branches (selection), recursion, pointers, etc. as well as abstraction and classes,
- The ability to determine the appropriate data structure for a purpose, and to implement it
- At least a basic understanding of time and space complexity,
- The willingness and ability to learn and gain experience in a new language, specifically Ruby, by implementing a program to meet requirements,
- At least a basic facility in using the individual VM assigned to you, git, and GitHub.

### Requirements:

Assume that you are a software engineer at GF Software Solutions, Inc., and that your company has been awarded a contract to develop a course management system for an educational institution. As part of that larger system, your current task is to develop a prototype that will help instructors with large courses with multiple sections to divide students into groups for classwork, based on different criteria or constraints.

Implement a prototype of an application, in Ruby, with the functionality described below.

- **Input data:** The program should allow the user to provide the name of a file that contains data in a specified format, read data from this file, and store them in an appropriate data structure for further processing.

The first line of the input file will identify the columns or the fields for the remaining rows. Each of the remaining rows will have the following fields separated by comma; one row represents data for one individual. A required field is one that cannot be empty; an optional field may or may not contain data (it can be empty). When a field is empty there will be a blank space between the two commas. Assume that the file will be formatted correctly.

*first\_name,last\_name,email,section,major1,major2,minor1,minor2*

- first\_name – first name; required
- last\_name – last name; required
- email – email address, including @ sign and domain; required
- section – the class section; required
- major1 – student's primary major; required
- major2 – 2<sup>nd</sup> major; optional
- minor1 – 1<sup>st</sup> minor; optional
- minor2 – 2<sup>nd</sup> minor; optional

## CSC 415 Fall 2020

- **Edit data:** The program should allow the user to add new students, delete students, and edit any of the information about students, e.g. update the second major.
- **Form groups:** The program should allow the user to enter constraints for groups, and then create groups that meet those constraints. For example, the user may specify that the class should be divided into groups with 5-6 students each, where all students are in the same section and in the same major. In this example, the 'section' constraint may be required to be satisfied (i.e. you cannot assign students from different sections into a group), while there may be flexibility in the 'major' constraint (so if there aren't enough students from a major to form a group, they may be placed with students from a different major). Another type of constraint may be that ideally all students in the group should be from different majors.

Allow the user the option of selecting different combinations of constraints. The constraints that would make sense for this program is your design decision.

- **List groups:** The program should display, neatly formatted, the groups, and the students in each group, along with section, majors, and minors. This will show whether the constraints specified have been satisfied.
- **Write groups to file:** The program should allow the user to specify the name of a file into which the information about the groups can be written.

The program must be hosted and demonstrated to work on the individual VM assigned to you, and should enable user-interaction, i.e. allow the user to select the tasks to be completed, specify names of files, etc. A sample input file will be posted on Canvas closer to the due date, but you should create a few of your own for development and testing purposes. You will be expected to demonstrate that the program works successfully with other files as well. You can assume that all input files will have the same format.

Use git for version control, and GitHub to host the remote repository. Add me as a collaborator to the repository on GitHub. Be sure to commit changes frequently so that you do not inadvertently lose any of your code.

The intent of the assignment is for you to demonstrate your ability to learn a new language (in this case, Ruby), and implement a solution to a problem using this language. Therefore,

- You cannot use any built-in algorithms, or algorithms / code you find from other sources.
- You must implement any searching or sorting algorithms you might need.
- You must give consideration to time and space complexity, use appropriate data structures, and implement efficient algorithms.
- You are encouraged to use functions from the built-in Ruby CSV library to read data from files, and write data out to files.

Follow good programming practices to ensure that your program is modular, secure, reliable and efficient. Use a good coding style with consistent indentation to ensure that your source code can be easily read. **Source code files must be documented with maintenance information.** You must also document your code well to explain functionality and design decisions. See the general guidelines for programs and documentation, posted on the Canvas page "Guidelines for Programs".

Useful resources to learn Ruby:

- **LinkedIn Learning** – Free with your TCNJ id and password; access it from the TCNJ Today page at <https://today.tcnj.edu/>.
- **codecademy.com** – a good place to start is <https://www.codecademy.com/learn/ruby>.
- <http://ruby-doc.org/> – Documentation for the Ruby programming language.

## CSC 415 Fall 2020

After the due date has passed, you must meet with me individually to demonstrate your application on the VM, and to discuss your program, to verify that it is complete and works correctly to meet the requirements. **Make an appointment using the link provided on the Canvas page for this assignment.** Each appointment slot is for 20 minutes though I expect that the actual time needed will be about 15 minutes; if you think you need more time, sign up for two consecutive blocks. You should come prepared to take notes on feedback provided.

### **Deliverables:**

On Canvas in the “**Assignment 1**” dropbox, submit a file called “readme.pdf” that specifies:

- Your VM name and password.
- The full file pathname for the program on your VM.
- The url for the GitHub repository.
- Instructions for running the program.
- Known bugs, issues or limitations.

On your VM, in an appropriate subdirectory:

- All the source code files.
- All the input and output files you created during development and testing.
- The file(s) I post on Canvas for final testing.

On GitHub, in an appropriate repository:

- All the source code files.
- All the input and output files you created during development and testing.
- The file(s) I post on Canvas for final testing.
- Instructions for running the program.
- Known bugs, issues or limitations.

**Label each source code file and the readme file with your name and assignment number.**