**CSCI165 Computer Science II**
**Lab Assignment: Group Work 10 Points**

Lab days will be spent working in groups. There will be a shorter group assignment followed by a longer individual assignment. **The group work is to be completed as a group** with 10 points of the overall lab grade dedicated to this effort. You will work in pairs (or maybe 3's since the class has an odd number of students and there are people who do not show up). Being absent does not excuse you from this assignment.

My goal with this arrangement is to allow you to develop problem solving skills without relying on the professor. This does not mean that I will not provide assistance or provide clarification when instructions are a bit murky; but I want you to collaborate with your classmates. I think you will find this arrangement enjoyable and beneficial despite the initial "ugh" moments of having to talk to other humans :)

**Submission:** Add source code files and any external resources to the group-work directory inside the lab directory. Push the changes to your remote repo.

**Groups:** These are generated randomly by a computer program and will change weekly

Group 1: Callisto Hess, Brendyn Weiss
Group 2: Justin Wade, Nicholas Parker
Group 3: Mehdi Nya Louhaichy, Hans Tang
Group 4: Garrett Davenport, Chauncey Smith
Group 5: Kaiya Hogg, Patrick Temple
Group 6: Justin Friends, Stephan Wagner
Group 7: Kaiyir Minoia, Michael Haupt, Augustus Phillips

**Objectives:**

- Use data types correctly, applying conversions when necessary
- Apply math operations
- Process String data using API methods
- Read/Write to external files

Travel Tickets Company sells tickets for airlines, tours, and other travel-related services. Because ticket agents frequently mistype long ticket numbers, Travel Tickets has asked you to write an application that filters out invalid ticket numbers. Ticket numbers are designed so that if you drop the last digit of the number, then divide the number by 7, the remainder of the division will be identical to the last dropped digit. This process is illustrated in the following example:

1. Examine a ticket number; for example, 123454.
2. Remove the last digit, leaving 12345.
3. Determine the remainder when the modified ticket number is divided by 7.
    1. In this case, 12345 divided by 7 leaves a remainder of 4 which does equal the original last digit. This ticket is valid.
4. Assign the Boolean value of the comparison between the remainder and the digit dropped from the ticket number.
5. Log valid ticket number to a file.
6. Get the next ticket number and repeat.

The program deals with data sets of 1000 potential ticket numbers at a time. The tickets can be anywhere from 5 to 8 digits long and you have no way of knowing how long a given ticket number will be. The goal of the problem is to filter out the valid ticket numbers from the invalid ticket numbers. The valid ticket numbers should be written to a file so that they can be distributed to customers.

You have been provided with some starter code for this lab. The starter code walks you through the String processing and math necessary to determine if a ticket number is valid.

Open the file ***Tickets.java*** and read the comments. They will direct you in your tasks. You will submit your work by pushing commits to your remote repository. Be sure to add a **.gitignore** file to the root of your local repository to include ***.class** files. I do not want class files to be submitted.

Rubric:

| **Requirement** | **Points** |
|---|---|
| Included Detailed Comments on provided code and included comments in additional code | 5 |
| Code is correct and identifies the appropriate ticket numbers | 5 |