

INTRODUCTION TO COMPUTER SCIENCE I

PROJECT 1

Scavenger Hunt!

Reminder regarding intellectual responsibility: This is an individual assignment, and the work you submit should be your own. Do not look at anyone else's code, and do not show anyone your code (except for me and the course TAs).

1 The game is afoot

A scavenger hunt awaits you! You will need to write a program that performs certain calculations (on year numbers) in order to follow the clues. The program's calculations are described below, instructions for writing the program are in Section 2.1, and the starting clue is then provided in Section 2.2. Note the map on the last page of this document—it divides the campus into a coordinate grid that will allow you to figure out where to go at each step.

The calculations: To perform this hunt, you will be finding clues in the form of 4-digit year numbers. These are (with the exception of the very first starting clue) years that you will find on buildings and statues on campus, so you can assume that they are between 1820 and 2018. Each year number will be an input to a program that performs an unnecessarily complicated calculation that yields coordinates. Using the attached map, you will go to those coordinates to find the next year-number clue.

Part of the calculation performed by this program depends on whether the given year is a *leap year*, as determined by these rules:

- If it is a multiple of 4, then it **is** a leap year, unless ...
- If it is a multiple of 100, then it **is not** a leap year, unless ...
- If it is a multiple of 400, then it **is** a leap year.

Therefore, 1924 was a leap year, but 1900 was not, yet 2000 was.

Once your program determines whether the given year is a leap year, it can then apply the correct one of the following two calculations to produce coordinates:

- If the year **is a leap year**, then:
 - The first coordinate is the absolute value of the difference between the first two digits and the last two digits of the number (e.g., if the year is 1908, the first two digits are “19” and the last two digits are “08”, so the first coordinate is $19 - 8 = 11$).
 - To obtain the second coordinate, first divide the year by 4. Then add 1 to the sum of the first and last digits of that result. Finally, divide by 2.

- If the year **is not a leap year**, then:
 - To obtain the **first coordinate**, first sum all the digits, and then:
 - * If that sum **is** a multiple of 7, then divide by 3 and add 1.
 - * If the sum **is not** a multiple of 7 **but is** greater than 10, then divide the sum by 3 and then add 7.
 - * If the sum **is not** a multiple of 7 **and is not** greater than 10, then add 7 to that sum.
 - To get the **second coordinate**, first check if the year number is divisible by its first two digits **or** the first digit is not divisible by the last digit. If either of those conditions holds, the second coordinate is one less than the result when the last 2 digits are divided by the first 2 digits. If neither condition holds, subtract the square of the first digit from the last two digits.

2 Your assignment

2.1 Getting started

Open a terminal, create a directory for this project, change into it, and grab some starting source code:

```
$ mkdir project-1
$ cd project-1
$ wget -nv --trust-server-names https://goo.gl/t6ZhNQ
$ emacs ScavengerHunt.java &
```

Complete the `ScavengerHunt` program so that it takes a year number as input, performs the calculation described in Section 1, and emits the coordinates that it calculates. When you run the program, your output should look something like:

```
[kgardner@romulus project1] $ java ScavengerHunt
Enter the year: 1908
11 6
```

If you check the map, you will find that those coordinates lead you to Charles Pratt Dormitory. As an additional test, the year 1933 should provide coordinates to Valentine Hall.

2.2 Hunt!

To begin, enter the year 3232. That should yield coordinates on the map, and you should go to that place. At each location there is a sign with a date on it (if there are multiple dates, use the **first** 4-digit year that appears). There are **six locations to find**, including this first one. At the **fifth location**, go into the building and find and write down a fun fact. At the **sixth and final location**, visit room 405, provide the fun fact you found at the previous location, and you will receive a prize.

You are **not** required to actually complete the hunt (after all, it's February and running around campus in the cold may not seem all that palatable). However, you **are** responsible for ensuring that your code works properly in all cases, so make sure you test it thoroughly! Actually doing the scavenger hunt will ensure that each case of your code gets tested at least once. Plus the prizes are fun!

3 Submitting your work

Submit your `ScavengerHunt.java` source code file with the CS submission system, using one of the two methods:

- **Web-based:** Visit the submission system web page.
- **Command-line based:** Use the `cssubmit` command at your shell prompt.

This assignment is due on Monday, March 5, 11:59 pm.

