

# CS 161 Program Two

## Purpose

The purpose of this program is to introduce the use of the **for** control structure to cause statements to repeat (iterate).

## Objectives

When you have completed this assignment, you will be able to

- design the solution to a problem that requires iteration;
- write a simple program that uses iteration;
- write assignment statements to perform simple computations; and
- maintain consistent formatting style to improve the readability of your program.

In addition, you will acquire some experience in debugging programs.

## Task

This program's job is to input daily snowfall amounts and output a report that shows each day's amount and reports the total snowfall for the week and the average daily snowfall.

Your job is to do the following:

- 1) figure out how to input seven values (one at a time) using a **for** structure, and accumulate a total as the values are entered;
- 2) figure out how to compute an average;
- 3) write the necessary additional code to complete the described job; and
- 4) document the program explaining what it does and who wrote it.

## Requirements for Reduced Credit (4-1/2 points)

The report your program generates should only show the snowfall for each day of the week, and one total and one average.

You should use no more than four variables.

The program must compile and it must show the correct values. Make sure that you don't truncate the average when it should show a fractional amount!

The output should be formatted as below.

The program must have documentation describing its purpose, the author, and the date it was written. Identifiers should be descriptive.

## Sample Output

```
Spout Springs
Weekly Snowfall Report
=====
```

```
Enter the snowfall for a day: 2
Day 1: 2
Enter the snowfall for a day: 5
Day 2: 5
Enter the snowfall for a day: 0
Day 3: 0
Enter the snowfall for a day: 0
Day 4: 0
Enter the snowfall for a day: 7
Day 5: 7
Enter the snowfall for a day: 1
Day 6: 1
Enter the snowfall for a day: 0
Day 7: 0
Total: 15
Avg: 2.14285714286
```

Note that at the moment there is no way to prevent the interfering lines of input statements.

### Additional Work for Full Credit (6 points)

To earn full credit for the assignment, satisfy the criteria shown above and address the following two extra criteria.

- 1) It would be nice if the prompt showed the day for which the user is entering the snowfall amount, as shown in the fragment below. Accomplish this *without* using any additional input statements.

```
Enter the snowfall for day 1: 2
Day 1: 2
Enter the snowfall for day 2: 5
Day 2: 5
Enter the snowfall for day 3: 0
Day 3: 0
```

- 2) Round the average to two decimal places (ie, it would report 2.14 for the above example and 3.25 for 3.2469217).

### For ½ point extra credit, solve this challenge:

- 3) Having the input prompts interspersed with the report lines is a nuisance. It would be nice if we could store the input values in a list and print the report after doing all the input. If you're bored, try this.