CST231 Spring 2025

Homework Week 7. Nested Loops

Objectives

- Create more flowcharts
- Use **nested loops**, in conjunction with if/else-if/else conditionals

Problem A. Finding Unique Factors (20 pts code, 10 pts flowchart)

https://codecheck.io/files/230316210692t39etbx6i6xxdisb0x0vktl

This program will print out and count the factors of each number in a given range. The number X is a factor of another number Y if we can evenly divide Y by X. For example, the number 8 has four factors: 1, 2, 4, and 8.

Your code should:

1. Allow the user to enter a starting number (must be greater than zero)
2. Allow the user to enter a stopping number (must be greater than the starting number)
3. Loop from starting number to stopping number, inclusive of both
4. For each number, print out all factors between 1 and the number

Instructions:

1. Create a flowchart to describe the algorithm
   (you do NOT need to include validating the starting and stopping numbers)
2. Write the code and test it

Sample output:

```
Enter a starting number (greater than 0): 1
Enter a stopping number (greater than 1): 10
Factors for 1: 1
1 has 1 factors

Factors for 2: 1 2
2 has 2 factors

Factors for 3: 1 3
3 has 2 factors

Factors for 4: 1 2 4
4 has 3 factors

Factors for 5: 1 5
5 has 2 factors

Factors for 6: 1 2 3 6
6 has 4 factors
```

```
Factors for 7: 1 7
7 has 2 factors

Factors for 8: 1 2 4 8
8 has 4 factors

Factors for 9: 1 3 9
9 has 3 factors

Factors for 10: 1 2 5 10
10 has 4 factors
```

You must complete the flowchart first (although your flowchart may not translate perfectly to your code:

1. Draw a complete flowchart
2. Download the flowchart as a PDF and upload to the course website as `factor.pdf`

When you have finished the problem correctly:

3. Download the report
4. Unzip the zip file
5. Rename report.html to `factor.html`
6. Upload to the course website

Incorrectly named files and wrong file types will not be graded.

Problem B. Summoner (10 pts)

https://codecheck.it/files/2212202427iil3rolnprpnimeisyl5i6s9

This program will count from 0 to a stopping number entered by the user. It will allow the user to choose whether or not to include each number in a sum. If the user says yes (y), then it will be added to the sum. If the user says no (n), it will not. Any other answer will cause the question to repeat. At the end, the program will print out the sum, plus a count of the numbers added to the sum.

Sample output from several runs:

```
What number should I count up to (1 or higher)? 5
Should I add the number 0 to the sum? (y/n) y
Should I add the number 1 to the sum? (y/n) y
Should I add the number 2 to the sum? (y/n) y
Should I add the number 3 to the sum? (y/n) fhgtfghj
Should I add the number 3 to the sum? (y/n) fghhfghfg
Should I add the number 3 to the sum? (y/n) hfghfg
Should I add the number 3 to the sum? (y/n) NO
Should I add the number 3 to the sum? (y/n) STOP
Should I add the number 3 to the sum? (y/n) y
Should I add the number 4 to the sum? (y/n) y
Should I add the number 5 to the sum? (y/n) y
You added 6 numbers for a sum of 15
```

```
What number should I count up to (1 or higher)? 7
Should I add the number 0 to the sum? (y/n) n
Should I add the number 1 to the sum? (y/n) y
Should I add the number 2 to the sum? (y/n) n
Should I add the number 3 to the sum? (y/n) y
Should I add the number 4 to the sum? (y/n) n
Should I add the number 5 to the sum? (y/n) y
Should I add the number 6 to the sum? (y/n) n
Should I add the number 7 to the sum? (y/n) y
You added 4 numbers for a sum of 16
```

When you have finished the problem correctly:

1. Download the report
2. Unzip the zip file
3. Rename report.html to `summoner.html`
4. Upload to the course website