Programming Project 02

This assignment is worth 10 points and must be **completed and turned in before 11:59 on Monday, January 29ᵗʰ 2018**

Assignment Overview
This assignment will give you experience on the use of loop and conditional statements. We are going work with the juggler sequence https://en.wikipedia.org/wiki/Juggler_sequence

Background
The juggler sequence is another of the well known sequences similar to the Collatz sequence. It has the following definition:

$$a_{k+1} = \begin{cases} \left\lfloor a_k^{\frac{1}{2}} \right\rfloor, & \text{if } a_k \text{ is even} \\\\ \left\lfloor a_k^{\frac{3}{2}} \right\rfloor, & \text{if } a_k \text{ is odd.} \end{cases}$$

The $\lfloor \rfloor$ symbol is the **floor** function, rounding any floating point value **down** to an integer. It is not rounding as you would typically define it. It is essentially truncation of the fractional part, leaving the integer part of the floating point number. It returns a floating point number.

- floor(123.23) is 123.0
- floor(123.999) is 123.0

Some things to note about the juggle sequence:
- Every known starting number eventually ends in the value 1. At that point the sequence ends.
- The numbers can get quite large, so make sure you use the correct type, a long. Even with the correct types you can exceed the size of a long integer.
  - if that occurs, you will see some strange results. Why is that?
- The `cmath` library provides a number of useful methods including `pow`, `sqrt` and `floor` (look them up).

Some example sequences are on the Wikipedia page above

Project Description / Specification
Your program takes as input three integer values (same line, space separated):
- the first two integers, the beginning and ending value in a range of integers you are going to examine, as space separated integers. They are in that order: the first is the lower range and the second is the upper range. Calculate the sequence **for each value** in that range ***inclusive*** of the endpoints.
- the third integer a value indicating whether to print each sequence: 1 (indicating yes) or 0 (indicating no).

Your program will print (look at the Mimir test cases for examples):
- the sequence for each number in the range on a separate line if the third input value is 1

- o note in printing the sequence the numbers are followed by a comma, except for the last number in the sequence. A personal pet peeve of mine.
- On a separate line, two comma separated numbers. For the sequence that is the **_longest_**, print the starting number and the length of the sequence (again, see Mimir for format)
- On a separate line, two comma separated numbers. For the sequence that contains the **_biggest number_**, print the starting number and the biggest number in that sequence.

Requirements
1. You should check that the first two input range numbers are >= 2, and that the smallest (the first entered value) is indeed strictly smaller than the second (the second value). If not, the program prints an error message and halts

Deliverables
  proj02/proj02.cpp . The name of the directory is proj02, the name file you turn in should be exactly proj02.cpp. Just like the lab, you must click on proj02, the directory under "Project 02 – juggler sequence ", to submit to Mimir. Not the file, the directory.

Notes
1. The **%** (modulus) operator is useful for this project.
2. If you ask for a large enough element, you might overflow an integer. If you go big enough, you will overflow a long (though it will take awhile).