Task 04-01

- Write a Python program called **lcm_gcd.py** to calculate the lowest common multiple (LCM) of two integers
- You may only use basic arithmetic operators - no looping constructs
- You may also leverage Python's built-in greatest common divisor (GCD) function
- Calculate and display the LCM of **447618** and **2011835**
- Upload your solution to the BNL QIS101 SharePoint site

Task 04-02

- Write a Python program called **herons_method.py** to implement **Heron's Method** for numerically estimating square roots
- Using the **built-in random** module, generate a random integer such that
- Using Heron's method, find such that with
- Display variables and with rounded to eight digits to the right of the decimal
- Upload your solution to the BNL QIS101 SharePoint site

Task 04-03

- Update **factor_quadratic.py** to properly factor this quadratic:

- For extra street cred (aka bonus points):
  - Factor out and display the GCD shared amongst J, K, and L
  - Strengthen the code to process ***negative*** coefficients
  - Stop after displaying the first found factor (if any)
  - Handle quadratics like
- Upload your solution to the BNL QIS101 SharePoint site

Task 04-04

- Add code to **main()** in **gen_continued_fractions.py** to display two GCFs that **I discovered** that approximate
- Upload your solution to the BNL QIS101 SharePoint site

GCF #1

$$\pi = 3 + \cfrac{1}{7 + \cfrac{1}{15 + \cfrac{25}{23 + \cfrac{65}{31 + \ddots}}}}$$

$$\pi = [\overset{a_0}{3};\ \overset{b_0}{1},\ \{(\overset{b_n}{8n^2-7})\,|\,(\overset{a_n}{8n-1})\}]$$

GCF #2

$$\pi = 2 + \cfrac{8}{6 + \cfrac{12}{10 + \cfrac{32}{14 + \cfrac{60}{18 + \ddots}}}}$$

$$\pi = [\overset{a_0}{2};\ \overset{b_0}{8},\ \{(\overset{b_n}{4n^2+8n})\,|\,(\overset{a_n}{4n+2})\}]$$

Did I beat Euler?