Project_Euler_012

February 4, 2018

1 Project Euler Problem 12

The sequence of triangle numbers is generated by adding the natural numbers. So the 7th triangle number would be 1 + 2 + 3 + 4 + 5 + 6 + 7 = 28. The first ten terms would be:

1, 3, 6, 10, 15, 21, 28, 36, 45, 55, ...

Let us list the factors of the first seven triangle numbers:

```
 1: 1
 3: 1,3
 6: 1,2,3,6
10: 1,2,5,10
15: 1,3,5,15
21: 1,3,7,21
28: 1,2,4,7,14,28
```

We can see that 28 is the first triangle number to have over five divisors.

What is the value of the first triangle number to have over five hundred divisors?

```
In [6]: # Note: I can probably make this more efficient.

        from math import sqrt
        triangle = 3
        i = 2

        while triangle < 100000000:
            i += 1
            triangle += i
            divisors = 0
            for j in range(1, int(sqrt(triangle) + 1)):
                if triangle % j == 0:
                    divisors += 2
            if divisors > 500:
                print("{} has {} divisors".format(triangle, divisors))
                break
```

```
76576500 has 576 divisors
```