```
Python 3.12.0 (v3.12.0:0fb18b02c8, Oct  2 2023, 09:45:56) [Clang 13.0.0 (clang-1300.0.29.30)]
on darwin
Type "help", "copyright", "credits" or "license()" for more information.

==== RESTART: /Users/jreid/Documents/JLR_dev_code/merrimack/CSC6013/test.py ====
Running P5_1.py to calculate the number of digits in the binary expansion/representation of a
positive Integer n.
Where n = 256 and n = 750.
The binary expansion/representation of 256 has 9 digits.
The binary expansion/representation of 750 has 10 digits.

Calculate time complexity using the Master Method
T(n) = T(n/2) + 1, T(1) = 1
a = 1, b = 2, f(n) = 1
n ** log 1 base 2 = 0
 1 > 0 since f(n) is faster rule 3 is applied.
 T(n) = O(f(n))
 stop point T(n) = 1 + (log n base 2)
O(log n)


Running P5_2.py o calculate the sum of the squares of the positive Integers 12 + 22 + 32 + …
+ n2, given the value of n.
Where n = 12 and n = 20.
The sum of the squares of 12 is 650.
The sum of the squares of 20 is 2870.

Calculate time complexity using Back Substitution
 T(n) = T(n-1) + n ** 2, T(1) = 1
 T(n) = (n(n + 1)(2n +1))/6
O(n)
```