Lecture 15: Classes, Inheritance, Access Control
```python
class Chart:
    def __init__(self, title):
        self.title = title

    def title(self):
        return self.title

    def __str__(self):
        return "{}".format(self.title)

class Histogram(Chart):
    def __init__(self, bins, title):
        self.bins = bins
        self.counts = [0]*len(self.bins)
        super().__init__(title)

    def index(self, bin):
        return self.bins.index(bin)

    def add_to_bin(self, bin, count):
        self.counts[self.index(bin)] += count

    def count(self, bin):
        return self.counts[self.index(bin)]

    def __str__(self):
        h = "\".join(["{}:{}\" for (x,y) in zip(self.bins,self.counts)])
        return "[{} \{}\".format(super().__str__(), h)
```
The sequence of triangle numbers is generated by adding the natural numbers. So the $7^{th}$ triangle number would be $1 + 2 + 3 + 4 + 5 + 6 + 7 = 28$. The first ten terms would be:

$$1, 3, 6, 10, 15, 21, 28, 36, 45, 55, \ldots$$

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?