Problem 44 - Pentagon Numbers

Gautam Manohar

16 July 2018

This document originally appeared as a blog post on my website. Find it at gautammanohar.com/euler/44.

1 Problem Statement

Pentagonal numbers are given by the formula

$$P_n = \frac{n(3n-1)}{2}.$$

The first ten pentagonal numbers are

$$1, 5, 12, 22, 35, 51, 70, 92, 117, 145, \ldots \tag{1}$$

It can be seen that $P_4 + P_7 = 22 + 70 = 92 = P_8$. Also, $P_7 - P_5 = 70 - 35 = 35 = P_5$ is also pentagonal.

Generalizing for a given k, find all P_n, where $n < N$, such that at least one of $P_n \pm P_{n-k}$ is pentagonal.

2 My Algorithm

Using the formula for P_n, we find a formula for its inverse:

$$\begin{aligned} 2P_n &= 3n^2 - n \\ n &= \frac{1 \pm \sqrt{1 + 4 \cdot 3 \cdot 2P_n}}{6}. \end{aligned} \tag{2}$$

Because n must be positive, we have that x is a pentagonal number if and only if

$$\frac{1 + \sqrt{24x + 1}}{6} \tag{3}$$

is an integer.

From here we use a brute force approach. For each $K < n < N$, test whether one of $P_n \pm P_{n-k}$ is pentagonal. This solution has time complexity $O(N - K)$.

2.1 Project Euler

The Project Euler version of this problem is more involved. We must instead find the first P_n such that both $P_n - P_{n-k}$ and $P_n + P_{n-k}$ are pentagonal. To do this, we check all $k < n$ for each n until we find the answer. I used the following code, along with the same `pent` and `is_pent` functions as in the HackerRank solution:

```
n = 0
while True:
    n += 1
    a = pent(n)
    for k in range(1,n):
        b = pent(n-k)
        if pent(a-b) and pent(a+b):
            print(a-b)
            break
```