FiveThirtyEight's Febuary 11, 2022 Riddler

Emma Knight

February 11, 2022

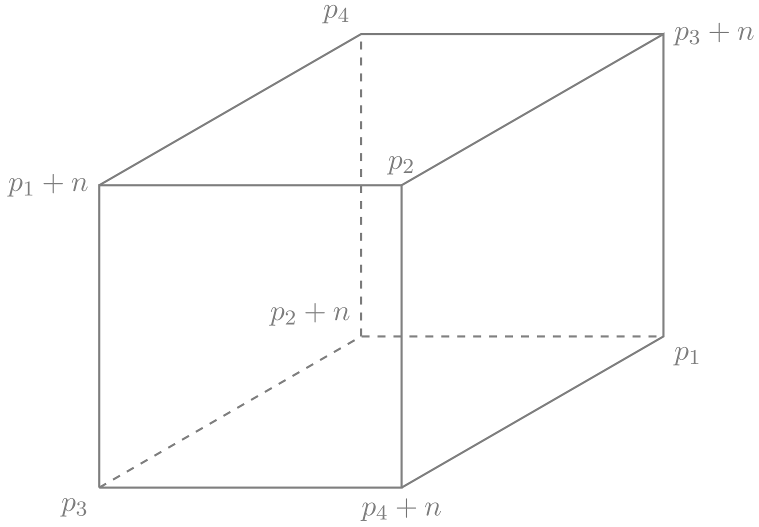
This week's riddler, courtesy of Dean Ballard, is about primes and cubes:

Question 1. *Consider a cube, which has eight vertices, or corners. Suppose I assign a prime number to each vertex. A "face sum" is the value I get when I add up all four prime numbers on one of the six faces.*

Can you find eight primes and arrange them on a cube so that the six face sums are all equal?

Extra credit*: Can you find another set of eight primes that can similarly be arranged on the vertices of a cube? How many more can you find?*

I will explain how to generate a lot of these cubes, and then give some examples. Choose prime numbers p_1, p_2, p_3, and p_4, as well as an integer n, such that $p_i + n$ is also prime for all i, and the eight primes generated this way are distinct. Then, one creates the following cube:



On this cube, every face sums to $p_1 + p_2 + p_3 + p_4 + 2n$. To get an example of this, take $p_1 = 3$, $p_2 = 11$, $p_3 = 17$, $p_4 = 29$, and $n = 2$. There are a *lot* of examples, and you are invited to do a brute force search to find as many as you please. To give context for just how many such cubes there are, looking through the first 15 odd primes (i.e. the primes up to 53), there are 173 such cubes (this counts two layouts that are symmetrical but aren't literally the same primes on the same verticies as the same cube). Adding the next prime into the list ups this number to 286[1], so it is not hard to find as many as you want using straight up brute force.

To see why this every such cube is of this form, notice that setting two adjacent faces to sum to the same number is the same thing as setting two opposite parellel sides to sum to the same number, and this is the same thing as setting two diagonal differences to be the same (suitably oriented). This gives that every diagonal difference is the same (again suitably oriented), and shows that every solution is of this form.

Finally, here is my python brute force code:

```
import itertools

primes = [3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61]

octuples = itertools.permutations(primes, 8)
valid = []

for l in octuples:
    if ((l[0] + l[1] == l[4] + l[5]) and
        (l[2] + l[3] == l[6] + l[7]) and
        (l[0] + l[3] == l[5] + l[6]) and
        (l[1] + l[2] == l[4] + l[7])):
        valid.append(l)
        print(l)

print(len(valid)//48)
```

[1] Adding in 61 as well gives 376 total cubes