August 2018 - Challenge

There is a well-known riddle about two friends (Alice and Bob) who went walking in the desert.
Alice had two gallons of water and Bob had three gallons.
They met Charlie who had no water at all and they all (Alice, Bob, and Charlie) shared the five gallons of water evenly.
As a token of his gratitude, Charlie gave them five gold coins.
What is the fair way to split these five coins?
Hint: it is *not* the obvious two to Alice and three to Bob. (Why?)

Our challenge this month is to solve a similar problem:
In J.R.R. Tolkien's works, nine rings were given to humans. Our challenge is this: Each one of the nine ring holders brought W_i droplets of water (W_1, W_2, W_3, ..., W_9).
Sauron (who had the one ring to rule them all) came without any water at all. They all shared their water; and each one of the ring holders got G_i (G_1, G_2, G_3, ..., G_9) tiny nuggets of gold in return. Where W_1+W_2+...+W_9 = G_1+G_2+...+G_9.
However, the 18 W_i and G_i water droplet and gold nugget amounts, respectively, are all different and at least 17 of them are prime numbers.
Find the nine W_i's.

Solution:

Hi Oded,

Thank you for your feedback. If you have the time, I have amended my submission.
first let total_Water = (W_1+ W_2,+W_3+ …+W_9) and total_Gold = (G_1+G_2+G_3+ …+G_9)

Since the ring holders and Sauron share the water, each ring holder and Sauron consume total_Water/10 droplets of water. So, for each ring holder to fairly receive G_i gold nuggets, we need

G_i = total_Gold*(W_i - total_Water/10) / (total_Water/10) using algebra and total_Water = total_Gold this equation simplifies to

G_i = (10*W_i - total_Water)

For the original Alice, Bob, and Charlie puzzle, this last equation reduces to

Alice_coins = (3*2 - 5) = 1 coin   Bob_coins = (3*3 - 5) = 4 coins

The minizinc constraint program below:

```
include "alldifferent.mzn";

include "globals.mzn";

array[1..18] of var 1..1001: x;

var int: totalw = sum(i in 1..9)(x[i]);

var int: totalg = sum(i in 10..18)(x[i]);

constraint totalw == totalg;

constraint forall(i in 1..9)(10*x[i]- totalw == x[i+9]);

constraint alldifferent(x);

%Filter first through small prime divisors

constraint forall(i in 1..17)( x[i] mod 2 != 0 /\ x[i] mod 3 != 0 /\ x[i] mod 5 != 0 /\ x[i] mod
7 != 0 /\x[i] mod 11 != 0 /\ x[i] mod 13 != 0 /\ x[i]>1);

solve satisfy;
```

Found the following corresponding W_i, G_i prime distinct pairs:

(223, 13) (227, 53) (229, 73) (233, 113) (239, 173) (241, 193) (251, 293) (257, 353) (317, 953)

where

total_Water = 223+227+229+233+239+241+251+ 257+317 = 2217

total_Gold = 13+53+73+113+173+193+293+353+953 = 2217
each G_i satisfies G_i = (10*W_i - total_Water) for example 953 = 10*317 - 2217. All consumed 2217/10 = 221 droplets of water, which is less than the number of droplets each ring holder brought. Each ring holder then receives a positive number of gold nuggets, ranging from 13 min and 953 max.

Thanks for considering,

Charles Joscelyne

Janet Joscelyne