BALANCE POWER

You can weigh all integer masses from 1 to 60 with 6 weights putting the weights in one pan and the object in the other pan. Which weights are used?

What about weighing all integer masses from 1 to 1000?

What about from 1 to n?

What could this have to do with the way computers work?

For a bigger challenge can you find how many weights are needed if you put weights in both pans?

Help

Explain why this list shows how to weigh masses 1, 2, 3, 4, 5, 6, and 7 units and ask learners to continue the list for masses 8, 9, 10 etc.

Weights

1

1 and 2

1 + 2 = **3**

1, 2 and 4

1 + 4 = **5**

2 + 4 = **6**

1 + 2 + 4 = **7**

1, 2, 4 and 8?

Extension

We use a decimal number system with units, tens, hundreds, thousands … etc (all powers of 10). But 10 is **not** the only number that can be used as the base for a number system. Other numbers can be used e.g. 2 or 3 or any other number. The ancient Babylonians had a base 60 number system and we still have traces of it with seconds and minutes. Computers use a base 2 number system.

The bigger challenge is to see how many weight you need to balance every each 1, 2, 3, … , 13 units using both pans. What pattern do you notice in the weights used? Can you explain it?

The weights can be coded:

- either as +1 times the weight when the weight is in the **opposite** pan to the object,
- or 0 times the weight when the weight is **not used** at all,
- or -1 times the weight when the weight is in the **same** pan as the object.

We are using a numbering system

$$a_0 + a_1x + a_2x^2 + a_3x^3 + \ldots$$

where the coefficients $a_0, a_1, a_2, a_3, \ldots$ are all 1, 0 or -1 and x=3.

This is equivalent to the base 3 number system.

All integers can be written in base 3 with digits 0, 1 and 2, that is as

$$b_0 + b_1x + b_2x^2 + b_3x^3 + \ldots$$

where the coefficients $b_0, b_1, b_2, b_3, \ldots$ are all 0, 1 or 2 and x=3. This explains why all objects weighing an integer number of units can be weighed with a set of weights that are all powers of 3.