

Binary Division

8 min

Binary division is the last fundamental arithmetic we will learn, as most other operations are built on these four functions.

Just like in multiplication, binary division follows the same algorithm as its sister operation in the decimal numbering system. Also just like in the decimal system, dividing by zero is meaningless, or for our sake, impossible.

Binary Numbers	Result
0 / 1	0
1 / 1	1
0 / 0	undefined
1 / 0	undefined

To show the similarities, we will use the same numbers from the equation in our previous exercise on multiplication: $1800 / 120$.

Alright, now let's work through all the steps of the process as we go about doing the same task in binary.

1. Set the problem up for long division. Make sure to put the divisor and dividend in the proper positions.
2. Find the first instance that the divisor can subtract out of the dividend.
3. Carry the next digit from the dividend down to the result, if the divisor can subtract into it, place a 1 in the quotient and subtract, if not, place a 0 in the quotient and carry the next digit down until you can subtract.
4. Repeat the process until you have carried down the last digit of the dividend.
5. If the result of the last subtraction is 0, then the problem is complete. Any leftover bits are considered the remainder, we will not do fractions and decimal division, even though the process is the same.

Instructions

1. Checkpoint 1 Passed

1.

Create a new variable, `divide_49_by_7` and set it equal to the binary result of $110001_2 / 111_2$

Hint

Take another look at this diagram to make sure you've set up your problem properly.

2. Checkpoint 2 Passed

2.

Create a new variable, `divide_728_by_8` and set it equal to the binary result of $1011011000_2 / 1000_2$

script.py

Your code below here:

```
divide_49_by_7 = 111
```

```
divide_728_by_8 = 1011011
```