Binary Multiplication

7 min

Now that the fundamentals of addition and subtraction are under our belt, multiplication and division will be a cinch.

The great part about binary multiplication is that you only need to know your multiplication table up to 1!

Binary Numbers	Result
0 * 0	0
0 * 1	0
1 * 0	0
1 * 1	1

Yep, it's really that easy!

Here is the long-form standard decimal multiplication problem, 120 x 15:

```
  120
 x 15
 ----
  600
+ 1200
 ----
 1800 <- Final answer
```

to Clipboard

Binary multiplication follows this exact same process, multiply and then add the results together. For larger multiplication problems you would repeat step 3 as many times as the bottom number is long.

Let's do the same problem, 120 x 15 except in binary.

1. Line up problem, larger number on top, place values aligned:

```
 1111000
 x  1111
 --------
```

to Clipboard

1. Multiply the top number by 1 of the LSB of the bottom number:

```
 1111000
 x  1111
 -------^
```

```
   1111000
```

1. Add a 0 to the next row and multiply all the top numbers by the next LSB (repeat as necessary):

```
  1111000
 x  1111
 ------^-
  1111000
 11110000
```

1. Repeat the same process for the next bit

```
   1111000
  x  1111
  -----^--
   1111000
  11110000
 111100000
```

1. Repeat the same process for the next bit

```
    1111000
   x  1111
   ----^---
    1111000
   11110000
  111100000
 1111000000
```

1. Add the results together

```
    1111000
   x  1111
    --------
    1111000 <- Add
   11110000 <- all
  111100000 <- these
 1111000000 <- together
  ----------
 11100001000 <- Final Answer
```

Instructions

1. Checkpoint 1 Passed

1.

Create a new variable multiply_12_and_6 and set it equal to the binary result of 1100_2 x 110_2

Hint

If you run into trouble, remember the addition rules as well:

- o 1 + 0 = 1
- o 1 + 1 = 10
- o 1 + 1 + 1 = 11

2. Checkpoint 2 Passed

2.

Create another variable multiply_50_and_15 and set it equal to the binary result of 110010_2 x 1111_2

script.py

```
# Your code below here:
multiply_12_and_6 =  1001000
multiply_50_and_15 = 1011101110
```