The Power of Binary

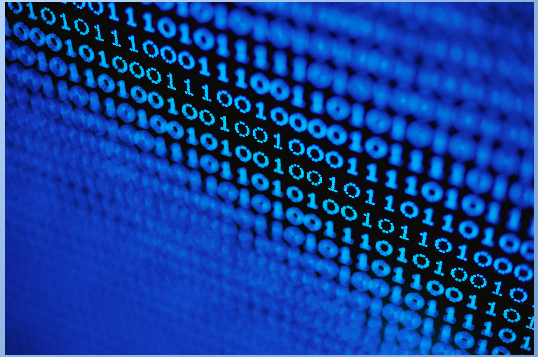
0, 1, 10, 11, 100, 101, 110, 111…

What is Binary?

- “a binary number is a number expressed in the binary numeral system, or base-2 numeral system, which represents numeric values using two different symbols: typically 0 (zero) and 1 (one)”
- “each digit is referred to as a bit”
- for example:
  0, 1, 10, 11, 100, 101, 110, 111 in the binary numeral system corresponds to 0, 1, 2, 3, 4, 5, 6, 7 in the decimal numeral system (or base-10 numeral system)

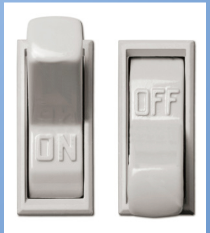
Why is Binary So Powerful?

- "because of it's straightforward implementation, the binary system is used internally by almost all modern computers and computer-based devices"
  - a binary digit or "bit is the basic unit of information in computing and digital communications"



So how do Computers use Bits?

- “in most modern computing devices, a bit is usually represented by an electrical voltage or current pulse, or by the electrical state of a flip-flop circuit”
- commonly, when the electrical voltage is high, the bit becomes a 1, and when the voltage is low, the bit becomes a 0
- you can also think of this as a switch that is a 1 when flipped on, and a 0 when turned off



Let’s Learn Some Binary!

- first, let’s break down the decimal numeral system (base-10) and use it to better understand the binary numeral system (base-2)
- in the base-10 system, we read eight-hundred forty-seven as 847
  - we can break this down to 800 + 40 + 7
  - even further gives us $8*10^2 + 4*10^1 + 7*10^0$

So how do Computers use Bits?

- the central processing unit (CPU), often referred to as the brain of the computer, is what holds millions, even billions of these “switches” or transistors
- through a series of steps, our computers break down the commands we give them into sequences of bits, essentially just a bunch of 1s and 0s. these sequences can be interpreted and evaluated by the CPU and GPU (graphics processing unit), and largely contribute to the results we see on our monitors

Let’s Learn Some Binary!

- we just converted 847 to $8*10^2 + 4*10^1 + 7*10^0$
- this is where the term “base-10” comes in
- each digit of a base-10 number corresponds to the digit itself multiplied by a power of 10, starting at a power of 0 at the rightmost digit and increasing by 1 each subsequent digit to the left

847

$$8*10^2 + 4*10^1 + 7*10^0$$

Let’s Learn Some Binary!

- one last thing to know about the base-10 system is that each digit can hold exactly 10 values
  - 0, 1, 2, 3, 4, 5, 6, 7, 8, 9

Let’s Learn Some Binary!

- using what we just learned about the decimal (base-10) system, let’s apply this to the binary (base-2) system
- first, because we are working with the base-2 system, each digit is restricted to the values 0 and 1
- also, we multiply each digit by a power of 2 this time instead of a power of 10

Let's Learn Some Binary!

- let's break down the binary number 10011
- using powers of 2, we get

$$10011$$

$$1*2^4 + 0*2^3 + 0*2^2 + 1*2^1 + 1*2^0$$

$$16 + 0 + 0 + 2 + 1$$

$$19$$

- so 10011 in binary translates to 19 in decimal!

Some Practice Problems

- okay, let’s take a couple minutes to convert these numbers from binary to decimal
- 11101
- 1010101

Some Practice Problems (Answers)

- okay, let's take a couple minutes to convert these numbers from binary to decimal
- 11101 29
- 1010101 85

A Binary to Decimal Trick

- instead of writing out all those powers of 2, you can write power of 2 equivalents above the digits, like this

64	32	16	8	4	2	1
1	0	1	0	1	0	1

- if the binary digit is 1, add the number above it, otherwise don't
- if you need to add another binary digit, simply multiply the leftmost top number by 2, and add it on the end (so 64 → 128)

Some More Practice Problems

- okay, let's try to convert a few more numbers from binary to decimal
- 10
- 111111
- 101010101

Some More Practice Problems (Answers)

- okay, let's try to convert a few more numbers from binary to decimal
- 10 2
- 111111 63
- 101010101 341

Converting Decimal to Binary

- the method for converting decimal to binary is a little trickier
- we’ll call it the “short division by 2 with remainder” method
- the idea is to continually divide the decimal number by 2 with remainder, until the number becomes 0
- then you group up all your remainders from last to first and construct what becomes the binary equivalent

Converting Decimal to Binary

- let’s give an example of the method

Convert 25 from decimal to binary.

$$
\begin{array}{rl}
2\,\overline{)25} = 12 & R1 \\
2\,\overline{)12} = 6 & R0 \\
2\,\overline{)6} = 3 & R0 \\
2\,\overline{)3} = 1 & R1 \\
2\,\overline{)1} = 0 & R1
\end{array}
\quad = \quad 11001
$$

Some Practice Problems

Convert 19 to binary.

Convert 62 to binary.

Some Practice Problems (Answers)

Convert 19 to binary. 10011

Convert 62 to binary. 111110