Hexadecimal numbers

Binary numbers are a great way for computers to represent numbers. They're not as great for humans though—they're so very long, and it takes a while to count up all those 111s and 000s. When computer scientists deal with numbers, they often use either the decimal system or the **hexadecimal system**. Yes, *another* number system!

Fortunately, number systems are more alike than they are different, and now that you've mastered decimal and binary, hexadecimal will hopefully make sense.

In the decimal system, each digit represents a power of 10—the ones' place, the tens' place, etc. We call 10 the **base** of the decimal system.

CHECK YOUR UNDERSTANDING

What is the base of the binary system? __________

Answer (2).

In the hexadecimal system, each digit represents a power of 16.

Here's how to count to 10 in hexadecimal: 1, 2, 3, 4, 5, 6, 7, 8, 9, A

That looks very familiar until the final "number" A . You see, in the hexadecimal system, each digit needs to represent the values 0-15, but the decimal numbers 10-15 don't fit into a single digit. To work around that, the hexadecimal system uses the letters A-F to represent the numbers 10-15.

Here, let's count from 10 to 15: A,B,C,D,E,F. Strange but true!

Let's count higher now. Here's the decimal number 24_{10}, represented in hexadecimal as 18_{16}:

1	8
16	1
16^1	16^0

This number requires two digits, where the right digit represent the ones' place (16^0) and the left digit represents the sixteens' place (16^1). The same way that we added decimal and binary numbers, we can add up $(1 \times 16)+(8 \times 1)$ and see that hexadecimal 18 equals the decimal value 24.

Let's try a number with a letter in it. Here's the decimal number 27, represented in hexadecimal as 1B:

1	B
16	1
16^1	16^0

To understand this one, we need to first remember that B represents the value 11. I do that by counting the letters on my fingers, but you can count however you'd like, as long as you remember to start with A representing 10.

Then we can add it up like before and see that $(1 \times 16)+(11 \times 1)$ does equal the decimal value 27.

CHECK YOUR UNDERSTANDING

Let's see if this is making sense. What decimal value is represented by the hexadecimal number 1F?

Answer (31).

Why base 16?

At this point, you might be wondering what it is that computer scientists like so much about the hexadecimal system. Why use a system where we have to use letters to represent numbers? A little detour into history will show us why...

Early computers used 4-bit architectures, which meant that they always processed bits in groups of 4. That's why we still write bits in groups of 4, like when we write 0111 to represent decimal 7 even though we could just write 111.

How many values can 4 bits represent? The lowest value is 0 (all 0s, 0000) and the highest value is 15 (all 1s, 1111), so 4 bits can represent 16 unique values. Aha, there's that number 16!

Each group of 4 bits in binary is a single digit in the hexadecimal system. That makes it really easy to convert binary numbers to hexadecimal numbers, and it makes it a natural fit for computers to use as well.

Converting binary to hexadecimal

Let's prove how well binary and hexadecimal get along by converting from binary to hexadecimal.

We'll start with a short binary number:

0010

That's 4 bits long, which means it maps to a single hexadecimal digit. There's a 0 in every place except the twos' place, so it equals the decimal number 2. The decimal and hexadecimal systems both represent the numbers 0-9 the same way, so 0010 is simply 2 in hexadecimal.

Let's try a longer binary number:

1001 1010 0110 1100

One approach would be to figure out what decimal number is represented by that long string of 1s and 0s, and then convert that to hexadecimal. That approach would work, but gosh, it seems like a lot of work for this long of a number.

The easier approach is to convert each group of 4 bits, one at a time.

Starting with the left-most group of 1001, that equals $(1\times8)+(1\times1)$, the decimal number 9. That number is less than 10, so 1001 is simply 9 in hexadecimal.

The next group is 1010, or $(1\times8)+(1\times2)$, the decimal value 10. That number requires a letter representation in hexadecimal, A.

The next group is 0110, or $(1\times4)+(1\times2)$ the decimal value 6. That number is the same in hexadecimal, 6.

The final group is 1100, or $(1\times8)+(1\times4)$, the decimal value 12. That number requires a letter representation in hexadecimal, C.

Our final hexadecimal result is 9A6C. We did that conversion without ever understanding what decimal number is represented. I'll reveal now that both $1001\ 1010\ \ 0110\ 1100_2$ and $9A6C_{16}$ equal the decimal number 39 532. That's an awfully big number, I'm glad we converted it one digit at a time.

CHECK YOUR UNDERSTANDING

Now you try it. What hexadecimal number does this binary number represent? 1011 1001 0001 1000

(Answer: B918)

Uses for hexadecimal

In this unit on how computers work, we're mostly going to be dealing with binary numbers. It's important to understand the hexadecimal system as well though, because they'll pop up throughout later units and just generally, in the life of a programmer.

As one example from my life, web developers use hexadecimal numbers to represent colours. We describe colours as a combination of three components: red, green, and blue. Each of those components can vary from 000 to 255. A color like blue can be written as `rgb(0, 0, 255)` or the more concise hexadecimal version, `#0000FF`. Using that notation, we can describe 16^6 unique colors—more than 16 million colors!

Now that you know about hexadecimal numbers, keep your eyes out for them. You'll often see them written with an `0x` in front, like `0x4F`, or you might just recognize their distinctive mix of `0-9` with `A-F`. If you spot any interesting uses for them in the wild, share your discovery with the class.