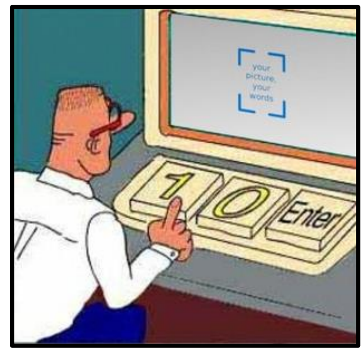
BINARY NUMBERING SYSTEM

WHAT IS THE BINARY SYSTEM?

The word "binary" describes a system that has only two possible digits. To understand this, let's first compare this to a system you're probably more familiar with, the DECIMAL system.



The word "decimal" describes a system that has ten possible digits. These are the digits 0 through 9. Every number expressed in the decimal system is a combination of these ten digits. You use the decimal system every day, it comes naturally, we all have 10 fingers and some of us use those 10 fingers to help with every day addition and subtraction. Counting using the decimal system (or in “base ten”) means that we count up to nine and then use double digits, like 10. 10 is not a single digit. It is a COMPOUND number. It means one ten plus zero ones. Likewise, the number 237 is actually 2 hundreds plus three tens plus seven ones. Each additional digit represents ones, the next is tens, then hundreds, then thousands, etc. so that it would look something like this:

	1,000,000 (10^6)	**100,000 (10^5)**	**10,000 (10^4)**	**1,000 (10^3)**	**100 (10^2)**	**10 (10^1)**	**1 (10^0)**
5							5
37						3	7
158					1	5	8

The binary system works essentially the same way, with the only difference that it only has two digits. These are visually expressed by the digits 0 and 1. Every number expressed in the binary system is a combination of these two digits.

WHY DO WE NEED THE BINARY SYSTEM?

The binary system is essential in technology. The reason is that a computer can only recognize two states, the presence of an electrical charge or the absence of an electrical charge. In other words, ON or OFF. A simple example is the light in your room. The switch has only two options, on or off. The binary numbering system is ideal for representing these two states because it consists of only two digits:

0 **Represents the absence of an electrical charge, or OFF**
1 **Represents the presence of an electrical charge, or ON**

Every communication that takes place inside your computer uses this binary system.

HOW DOES BINARY WORK?

If you're not used to them, binary numbers look pretty strange. Here's an example:

```
10101110
```

So what is this number in the decimal system? Converting binary numbers to decimal numbers is not that difficult if you know the secret.

THE SECRET OF THE BINARY SYSTEM

The first thing to know is that you read binary numbers from right to left. The second thing you need to understand is that each digit is based on a power of the number 2. Check this out:

```
2 to the power of 0 equals 1
2 to the power of 1 equals 2
2 to the power of 2 equals 4
2 to the power of 3 equals 8
2 to the power of 4 equals 16
2 to the power of 5 equals 32
2 to the power of 6 equals 64
2 to the power of 7 equals 128
…
```

See the pattern? Now let's take these numbers and put them in a template similar to the one we used above for the decimal numbering system. REMEMBER: binary numbers are read from right to left:

128 (2^7)	**64 (2^6)**	**32 (2^5)**	**16 (2^4)**	**8 (2^3)**	**4 (2^2)**	**2 (2^1)**	**1 (2^0)**

Now let's use this template on that ugly binary number from our earlier example.

128 (2^7)	**64 (2^6)**	**32 (2^5)**	**16 (2^4)**	**8 (2^3)**	**4 (2^2)**	**2 (2^1)**	**1 (2^0)**
1	0	1	0	1	1	1	0

Now we use simple multiplication and addition. If the binary number is a 1, it means this digit is ON or TRUE and we add the corresponding number from the template. If it is a 0, it means the digit is OFF or FALSE, and we do not add the corresponding number from the template.

In our example, the digits for 128, 32, 8, 4 and 2 are true, so we add:

$$128 + 32 + 8 + 4 + 2 = 174$$

This means our binary number 10101110 is the number 174 in the decimal system.

HOW DO YOU CONVERT DECIMAL NUMBERS TO BINARY?

As you saw in the chart we used to illustrate how the DECIMAL numbering systems works, any number can be represented by a base of 10. In BINARY, we simply use a base of 2 instead. The right most number in a binary code series indicates the number of ones, the next digit to the left indicates the number of twos, the next the number of fours, then eights, then sixteens, and so on doubling each time. Using the decimal numbers we used in the above chart, we could represent them in binary code as follows:

	128 (2^7)	64 (2^6)	32 (2^5)	16 (2^4)	8 (2^3)	4 (2^2)	2 (2^1)	1 (2^0)
5		0	0	0	0	1	0	1
37		0	1	0	0	1	0	1
158	1	0	0	1	1	1	1	0

To convert decimal to binary, we first find the nearest power of 2 to the decimal number, then we subtract it to the number. We do the same with the result of the subtraction and so on until the result is 0.

In the above example, to convert the number 158 into binary we do the following:

$$158 - 128\ (2^7) = 30$$
$$30 - 16\ (2^4) = 14$$
$$14 - 8\ (2^3) = 6$$
$$6 - 4\ (2^2) = 2$$
$$2 - 2\ (2^1) = 0$$

We put a 1 for each power of 2 used during the calculation process and 0 for each one not used.

BITS VERSUS BYTES

Ok, now that we understand the binary system and how binary numbers work, let's take a closer look at how your computer uses this system. Earlier we talked about 1 and 0 being **b**inary dig**it**s. That's an ugly phrase to use, so let's abbreviate it by combining the two words into a nice short term: Bit. Sound familiar? Thought so.

In our earlier example, we used the binary number 10101110. This number has 8 binary digits, or 8 bits. This is not a coincidence, because if you take a group of 8 bits, you have a byte. The reason we have to group bits is that if you are being fed a continuous stream of bits you have no idea where one piece of information ends and the next one starts. But if you receive groups of defined length (e.g. a byte containing 8 bits) it's easy to interpret them. A good example is how ASCII code (American Standard Code for Information Interchange) works. Any character you type on your keyboard is interpreted by your computer as a byte, an 8 digit binary number. For example, the letter "A" is expressed as the ASCII code 65. But 65 is a decimal number, so if you convert it to a binary number, you get 01000001. These 8 digits, or one byte, are known to your computer as the letter "A".

NOTE: This handout was adapted from material available at http://www.pcnineoneone.com/howto/binary1.html