Systems Organisation

Illustrating the Size Limitation of an Integer

32 bits

He made a C function to increment an integer that's a few thousand below 2^{32}.

When the program reaches (2^{32} - 1), $ffffffff_{16}$, the next increment takes it back around to 0

Adding 20_{10} to 4294967290_{10} gives 14_{10} as the number wraps around.

Note we're using unsigned long integers, so we have only positive numbers.

8 bits

With 8 bits we can represent $2^8 = 256_{10}$ things. 0 → 255.

A single byte is sufficient to represent characters in a western keyboard.

He's ignoring accented characters for the moment. There are character sets which use 16 bits.

ASCII

In the ASCII *character set* (American System for Coded Information Interchange), each binary value between 0 and 127 (or between 0 and 255 in extended ASCII) is assigned to mean a specific character.

Most computers make use of an extended set to access the full range of representations.

The upper 128 numbers (128 → 255) are used for accented characters in non-english languages.

If you type a letter on a keyboard, the keyboard sends the binary code for the letter you've pressed (e.g. 0V,0V,5V,0V,0V,0V,5V,5V) to the computer, and the computer then interprets that and takes that to be a certain value.

This value is then sent to the screen, the screen display finds the symbol in the ASCII table under that value.

For numbers, the keyboard sends I think the value of the number rather than the ASCII code.

For multi-digit numbers it's a bit more complicated (e.g. calculator) because the machine has to know that some of the numbers are in the 10s column, or the 100s column, and multiply them by the relevant power of 10.

He wrote a C program to take single character input from the keyboard and give the ASCII value for that character back to us.
Also showed us that adding characters in C is possible, e.g. adding "%" to "&" to get "K".

He made a new .txt document, wrote "Hi" in it, and then looked at the properties. The size was 2 bytes (1 for each character), and the size on disk was 4.00 KB.
The size on disk is large because it's allowing for the file to get bigger.
Hitting enter at the end of the file and saving again adds another 2 bytes, where one is a "newline" character, and the other is a "carriage return".

He did the same with a Word document, and checked the size, and it was 9.73 KB in size, because of all the extra formatting options included.