**CS50 – Introduction to Computer Science**

Lecture 0

Computer programming is about problem solving

The computer will not do what you want unless you are correct, precise and methodical

**Bit:** A binary digit, 0 or 1

These work well for a computer as it is equivalent to on or off. Either allow some electricity to flow or don’t. Transistors are the key component inside computers.

Using bits, numbers are trivially represented using the binary system. 8 bits are typically used to begin with, with preceding zeroes. A group of 8 bits is a **byte**.

Representing letters is done by representing each as a different number. For instance, ‘A’ is typically represented as the binary form of 65. ASCII – American Standard Code for Information Interchange. Computers all know the ASCII mapping. There is now a superset of ASCII, called Unicode. This will include representations of emojis, characters with accents etc. Unicode may use 8 bits for backwards compatibility or many more bits.

How then does a computer know whether is is dealing with a letter or a number: ‘A’ or ‘65’?

This is context dependent. Photoshop will know to look for colours, word processors will choose letters etc. Eg. Sound is often stored in the MIDI file format.

**Text Processor:** 72 73 33 → HI!

**Photoshop:** 72 73 33 → Some sort of yellowy colour

A pixel will have 8 bits for each colour in RGB, so 24 bits for one pixel

The essence of computer science is the conversion of inputs to outputs, via an **algorithm**. It is important to take human ideas and intuition and express them succinctly and precisely in programming languages.

**Pseudocode** is a way of expressing yourself to get an algorithm across

Programming languages have some commonalities:

**Functions:** Actions or verbs that solve some smaller problem

**Conditionals:**  ‘Forks in the road’, asking some sort of ‘if’ question

**Booleans:** True of False statements

**Loops:** Structures that allow for repetition