Programming languages

Machine code:

```
000010 00000 00000 00000 10000 000000
100011 00011 01000 00000 00001 000100
000000 00001 00010 00110 00000 100000
```

Assembly code:

```
MOVE R2, 0
ADD R2, R0, R2
SUB R1, R1, 1
```

High-level language:

```
pWord = input("Enter the password").upper()
```

Programming languages

There are two different types of programming language:

- **Low-level languages** - simple

- **High-level languages** - complex, closer to written English

Programming languages

Machine code:

```
000010 00000 00000 00000 10000 000000
100011 00011 01000 00000 00001 000100
000000 00001 00010 00110 00000 100000
```

- low-level language at a machine level

- the binary code that the CPU actually processes

- everything ends up as machine code - any other language needs to be translated to it anyway

- very difficult for humans to code in

- different processors have different sets of machine code

Programming languages

Assembly code:

```
MOVE R2, 0
ADD R2, R0, R2
SUB R1, R1, 1
```

- low-level language

- simple but limited set of instructions

- used to program embedded systems and hardware components directly

- 1:1 correspondence with machine code

- quick to translate into machine code

Programming languages

High-level language:

```
pWord = input("Enter the password").upper()
```

- for example, Python, Java, C# etc…
- most programs written in high-level languages
- more powerful
- closer to written English; easier and quicker to use
- has to be translated into machine code

Programming languages

High level languages:

Such as Python, Java etc...

- can be used on many different systems
- much more powerful set of constructs (selection, repetition, arrays etc…)
- closer to our languages - easier and quicker to learn, use, debug etc...
- need to be converted to machine code - takes CPU time and less efficient to run

Programming languages

Key points:

- Assembly Language and Machine Code are low-level languages
- most programs are written in high-level languages (e.g. Python)
- everything ends up as machine code
- there are advantages to using low-level languages