Machine Instructions

4 min

What the Instruction Set Architecture is centrally focused on is defining the machine instructions that our hardware can understand.

Machine instructions, or

Preview: Docs Loading link description

[binary](#)

 code, come packaged in very specific ways. If the software generates binary that doesn't follow the rules set out by the ISA, the hardware will fail in its processing of the data.

The way instructions are formatted is different from one architecture to the next. A *Complex Instruction Set Computer* (CISC) and a *Reduced Instruction Set Computer* (RISC) will not understand the same data.

An example machine instruction might look like this:

- 101100100111010011001001110000001

One

Preview: Docs Loading link description

[CPU](#)

 might interpret this as an instruction to add together two numbers, another might think it as a request to logically skip the next instruction if the result is true. A different CPU might not be capable of handling the data at all and return an error.

One way to quickly identify what type of computer a piece of

Preview: Docs Loading link description

[machine code](#)

 belongs to is to look at the length of the instructions. Typically, RISC computers use machine code that is all the same length, while CISC instructions can range in size from quite small all the way to 15 bytes (120 bits)!

Computers are only as intelligent as the rules we use to design and program them. Like most things in life, they follow the garbage in, garbage out principle, that is why so much time and research has been spent creating robust Instruction Set Architectures.

Instructions

1. Checkpoint 1 Passed

1.

Look at machine1 and the set of instructions assigned to it in the code editor. Each instruction is separated by a space. Which ISA type does code the instructions in machine1 belong to?

Show your guess by changing the value of answer1 from 'unknown'to either'RISC'or'CISC'`.

Hint

Remember, RISC computers typically use instructions that are fixed in length.

2. Checkpoint 2 Passed

2.

Look at the instructions for machine2 and change answer2 from 'unknown' to 'RISC' or 'CISC' based on the type of instructions these may be.

Hint

Computer instructions that vary in length are a telltale sign of a CISC computer.

script.py

Below are two sample machine instructions, try and guess which code belongs to which type of Instruction Set Architecture.

```
machine1 = '0101110011100011 0101001110100101 0011101001010101 1110101101001001 1100010001001001 0010010100100000 1111110001010010 0000000101001010'


# Part 1

answer1 = 'RISC'

print(f'Machine 1: I am an {answer1} computer!')


machine2 = '010111001110001101001010 0101001110100101 00111010010101010111001101001010101010101 11101011010010010101011111 11000100010010011100001001010101 0010010100100000101010001001001001111010010011101001001010010100001111100101001000 1000100100101 1111110001010010110100100010100101110010101 0000000101001010101011111'


# Part 2

answer2 = 'CISC'

print(f'Machine 2: I am an {answer2} computer!')
```