Low-level language characteristics 1

Practice 1

A program written in a low-level language is described as non-portable. What does that mean?

- ○ The program can only run on one machine
- ○ The program cannot be saved to a disc
- ○ The program cannot be copied
- ○ The program can only run on the processor type it was written for

Low-level advantage 1

Challenge 1

Which of these is a **advantage** of a low-level language compared to a high-level language?

- They are easier to write programs with
- They can directly address core hardware components
- They are suited to particular problems
- They are easier for humans to understand

Quiz:
STEM SMART Computer Science Week 39 (LMC)

Assembly language characteristics 1

Challenge 1

Which **two** statements are **disadvantages** of assembly languages, when compared to high-level languages?

- ☐ The programmer cannot add comments to their code
- ☐ Many lines of code are required to write complex programs
- ☐ Translated programs are not portable between computers with different architectures
- ☐ There is no way to implement selection or iteration statements

Quiz:
STEM SMART Computer Science Week 39 (LMC)

Direct addressing

Challenge 1

The following diagram shows the format of a machine code instruction:

Basic Operation				Addressing mode	Operand				
0	0	1	1	0	0	1	0	0	1
ADD					$01001_2 = 9_{10}$				

How many different memory locations can a programmer access using **direct addressing**?

Quiz:
STEM SMART Computer Science Week 39 (LMC)

Addressing modes

Challenge 1

🟦 ⬜

Tony has invented a new assembly language. An instruction in the language is structured as follows:

`Opcode, Addressing mode, Operand`

Each of the three parts is made up from three bits:

- The `LOAD` operation's code is `010`.
- The possible addressing modes are direct, immediate, and indirect. The code for direct addressing is `001`, for immediate addressing is `010`, and for indirect addressing is `011`.

The part of the main memory that Tony uses looks like this:

Address	Contents
000	010
001	Black Panther
010	Black Widow
011	101
100	Captain Marvel
101	Captain America
110	Thor
111	Hulk

Part A What data will be loaded? 1

What data will be loaded by the instruction `010 001 001`?

Part B What data will be loaded? 2

What data will be loaded with the instruction `010 010 011`?

Part C Load data

What would be the instruction to load the data 'Black Widow' using indirect addressing mode? Your answer should be a 9 bit binary number.

Quiz:
STEM SMART Computer Science Week 39 (LMC)

Bits for addressing mode

Challenge 1

In a 32-bit machine code instruction, 10 bits are used for the fundamental operation (e.g. add), 2 bits are used for the addressing mode, and 20 bits are used for the operand(s). How many different addressing modes can be supported with this structure?

- ○ 2
- ○ 4
- ○ 8
- ○ 10
- ○ 20

Quiz:
STEM SMART Computer Science Week 39 (LMC)

Machine code representation

Practice 1

●●

Machine code represented as binary is really difficult for humans to read and write. Programmers use hexadecimal to represent binary numbers instead. What is the hexadecimal representation of this machine code instruction?

```
0001 0000 0000 1101
```

Quiz:
STEM SMART Computer Science Week 39 (LMC)

Trace assembly code AQA style 1

Challenge 1

Karishma has written a program in assembly language and has given it to her classmates to try out. Trace the program and work out what the **final value of R2** will be at the end of the program.

It may be useful to write down the values of R0, R1 and R2 on paper whilst tracing the program.

Pseudocode

```
MOV R0, #8
MOV R1, #2
MOV R2, #10
start:
        CMP R0, #0
        BEQ end
        SUB R0, R0, R1
        ADD R2, R2, R0
        B start
end:
        HALT
```

- ○ 20
- ○ 30
- ○ 22
- ○ 38