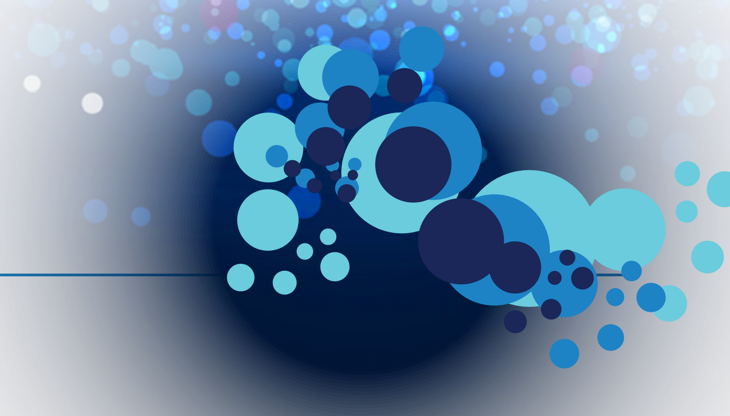
# High level code and machine code

Teacher's Notes



## Lesson Plan

| | | | | |
|---|---|---|---|---|
| Length | 60 mins | Specification Link | 2.1.7/cde | Programming languages |
| Learning objective | Students should be able to<br>(a) explain the difference between high level code and machine code<br>(b) explain the need for translators to convert high level code to machine code<br>(c) describe the characteristics of an assembler, a compiler and an interpreter | | | |

| Time (min) | Activity | Further Notes |
|---|---|---|
| 5 | Ask the students to quickly write down the names of any computer programming languages that they know.<br>Ask selected students for their ideas and ask if they know the difference between the terms 'low level language' and 'high level language'.<br>Show students the link to the specification. Explain the purpose and objectives of the lesson.<br>Stress that the methods shown used sound, light and the fixed positions of flags to encode letters which are then decoded by the recipient.<br>Computers use binary digits to do the same thing.<br>Show students the link to the specification. Explain the purpose and objectives of the lesson. | Other examples are available on YouTube. |
| 15 | Watch the set of videos. | |
| 5 | Ask some questions about the video to assess learning. e.g.<br>• What is the only language that a computer understands?<br>• Why don't programmers usually write programs using machine code?<br>• Name the types of translators that are used to convert other languages to machine code. | Machine code.<br>Difficult to visualize a program that is just a stream of 1s and 0s.<br>Assemblers, compilers, interpreters. |
| 20 | **Worksheet 1**<br>Students to complete Worksheet 1 either on paper or on computer using knowledge gained from video and internet research.<br>Ask individual students for their answers and discuss with the class so that all students will have the correct answers entered on their worksheets. | Answers provided. |

| Time (min) | Activity | Further Notes |
|---|---|---|
| 10 | **Activity**<br>Explain to the students that they are now going to use Assembly language in an on-screen simulator.<br>(http://www.cambridgegcsecomputing.org/weblink19)<br>Explain that they will be using mnemonics to input two numbers which will then be added together.<br>Explain that they will use the mnemonics:<br>INP – input<br>STA – store a number in a memory address<br>ADD – add numbers in accumulator<br>OUT – output the results<br>HLT – halt the program<br>The students should now use the demonstration run-through to carry out this activity.<br>Ask the students to carry out research to find the mnemonics for further actions and carry out other examples on the website. | |
| | **Extension Challenge/Homework**<br>The students should complete and submit **Worksheet 2**. | |
| 5 | **Plenary**<br>Remind students of the learning points to take away, based on those discussed earlier e.g.<br>• What is the only language that a computer understands?<br>• Why don't programmers usually write programs using machine code?<br>• Name the types of translators that are used to convert other languages to machine code and the differences between them. | Machine code.<br>Difficult to visualize a program that is just a stream of 1s and 0s.<br>Assemblers, compilers, interpreters. |

## WORKSHEET 1 ANSWERS

1 **Explain what is meant by 'machine code'.**

Machine code, also called machine language, is a computer language that is directly understandable by a computer's CPU (central processing unit).

It is the only language that a computer can understand.

The instructions – opcodes and operands - are represented by strings of bits – 1s and 0s.

2 **Explain why other languages were developed to allow humans to program computers.**

Writing and reading a program as a string of 1s and 0s is very difficult for a human.

It is difficult to memorise all of the instructions as 1s and 0s.

The speed of writing, testing and debugging a program  is very slow in machine language.

3 **(a) Why is 'assembly language' referred to as a 'low level' language?**

The instructions are very similar to those in machine language – there is usually a one-to-one relationship between the assembly and machine code instructions.

**(b) What are mnemonics and how are they used in assembly language?**

Mnemonics are tools that are designed to help a person's memory. They can be rhymes or phrases e.g. 'Richard of York gave battle in vain' used to help remember the order of the colours of the spectrum – red, orange, yellow, green, blue, indigo and violet.

In assembly language the mnemonics are English-like words used to represent the binary opcodes of machine language e.g. the mnemonic ADD is used in assembly language for the opcode 110001011.

SUB, MUL, DIV, RLC and RAL are other mnemonics used in assembly language.

**(c) What is an 'assembler' used for?**

An assembler translates the symbolic codes (mnemonics) of programs of an assembly language into machine language instructions.

It is translated to the machine code version in the ratio of one symbolic instruction to one machine code instruction.

## WORKSHEET 1 ANSWERS

4

### (a) What are ‘high level languages’?

High level languages are called ‘high-level’ because they are closer to human languages and are further removed from machine languages than assembly language.

There is no one-to-one relationship between the instructions in a high level language and machine language as there is with assembly language.

### (b) List *three* examples of a high level language.

Basic, C, Fortran, Python, Ada etc.

### (c) List three advantages of assembly language over a high level language.

It requires less memory and execution time.

It allows hardware-specific complex jobs in an easier way.

It is suitable for time-critical jobs.

It is most suitable for writing interrupt service routines and other memory resident programs.

### (d) List three advantages of using a high level language over assembly language.

Faster program development – it is less time consuming to write and then test the program.

It is not necessary to remember the registers of the CPU and mnemonic instructions.

Portability of a program from one machine to other. Each assembly language is specific to a particular type of CPU, but most high-level programming languages are generally portable across multiple architectures.

## WORKSHEET 1 ANSWERS

### 5 Compilers and interpreters are needed to translate the high level code into machine code. What is the difference between a compiler and an interpreter? Give an advantage and disadvantage of each.

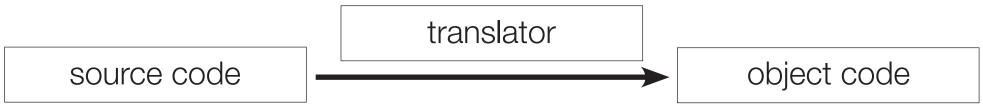
A compiler reads the whole high level code and translates it into a complete machine code program which is output as a new file and can be saved.
The biggest advantage of this is that the translation is done once only and as a separate process. The program that is run is already translated into machine code so is much faster in execution. The disadvantage is that you cannot change the program without going back to the original source code, editing that and recompiling.

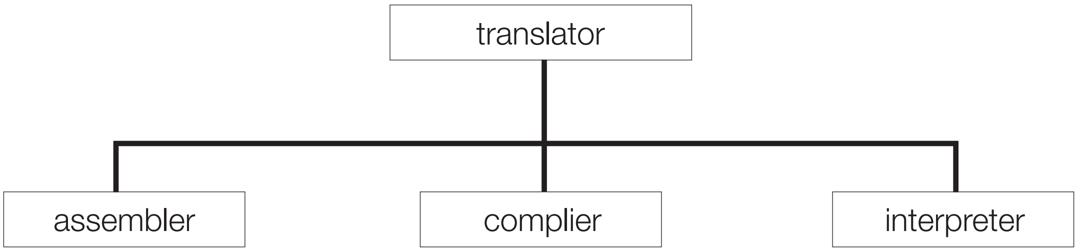
An interpreter reads the source code one instruction or line at a time, converts this line into machine code and executes it. The machine code is then discarded and the next line is read. The advantage of this is it's simple and you can interrupt it while it is running, change the program and either continue or start again.
The disadvantage is that every line has to be translated every time it is executed, even if it is executed many times as the program runs. And because of this interpreters tend to be slow.

### 6 Complete the following diagrams by labelling the boxes.

You may use the following words:

**source code, interpreter, assembler, object code, compiler**





## WORKSHEET 2 ANSWERS

1 **A consultant is developing a program for one of her customers.**
**The program is written in high level code and then translated to machine code.**

**(a) Describe two differences between high level code and machine code. (4)**

In high level code instructions use words.

In machine code instructions are in binary code.

High-level code is designed to be read by human programmers.

Machine code is to be read/executed by the computer

High level code can be portable/translated for different machines

Machine code is specific to a particular machine.

**(b) One type of translator which can be used is an interpreter.**
**Describe how an interpreter translates the high-level code to machine code. (2)**

Translates one line of HL code at a time…

… and executes it

… stops when it finds an error

… can be resumed

**(c) State the name of a different type of translator which can be used to translate high-level code to machine code. (1)**

Compiler