# Lesson 2: Activity 3

### Worksheet 1 - Compilers

Produce a graphics organiser to compare the differences between Low Level Languages and High Level Languages with similarities in the middle.

A compiler is a language translator that transforms [source code](http://en.wikipedia.org/wiki/Source_code) written in a [programming language](http://en.wikipedia.org/wiki/Programming_language) into machine code (also known as object code). A compiler produces this executable program (Object code)

Machine code consists of 0’s and 1’s and is directly understandable by the CPU. Therefore High Level code must be translated into Low Level code so that the computer can understand and execute it.

A compiler will go through the entire source code in one go, translating the entire source code at once. It will optimise the code as it does so.

If a compiler encounters errors, it produces an error report that it outputs along with the object code.

Compiled code tends to execute much quicker than interpreted code

Compilers tend to be used once a program development is finished so that the program can then be shipped out to customers and no compiler is needed during run-time. Customers therefore will not have access to the source code.

### Task

1. Read through the above text explaining what a Compiler is – underline/highlight key aspects
2. Produce a summary sheet of the text using at maximum 10 words – the rest of the summary sheet must be made up of pictures/diagrams/symbols/numbers/initials

### Worksheet 2: Interpreters

An interpreter analyses and executes each line of a high-level language program without looking at the entire program. Execution will be slower than for the equivalent compiled code as the source code is analysed statement by statement (line by line) during execution.

As each statement is analysed the interpreter calls routines to carry out each instruction. No object code is generated. This means the program has to be interpreted each time it is run.

The advantages of interpreters over compilers are that a program can be executed immediately without having to wait for it to be compiled. If a compiler and an interpreter exist for a high-level language, a programmer may use the interpreter to test sections during development

When an interpreter is working – if it encounters an error it will stop the interpreting and flag up the error to the programmer.

### Task

1. Read through the above text explaining what an Interpreter is – underline/highlight key aspects

Produce a summary sheet of the text using at maximum 10 words – the rest of the summary sheet must be made up of pictures/diagrams/symbols/numbers/initials

### Example to get you started

A compiler is a language translator that transforms source code written in a programming language into machine code

**Could be summarised with sketches of the following:**

Transformer logo – bottle of tomato sauce – few wiggly lines (represent code) – an arrow pointing towards a machine – followed by code.

**Example to get you started:**

each line of a high-level language program without looking at the entire program

**Could be summarised with sketches of the following:**

Several wiggly lines drawn (to represent code) with arrows pointing to each of these – a tall building drawn circling the High Level – a square to represent the program drawn – a magnifying glass focussed on one part of the square.

We’d like to know your view on the resources we produce. By clicking on ‘[Like](mailto:resources.feedback@ocr.org.uk?subject=I%20liked%20this%20GCSE%20(9-1)%20Computer%20Science%20Lesson%202%20Activity%203)’ or ‘[Dislike](mailto:resources.feedback@ocr.org.uk?subject=I%20disliked%20this%20GCSE%20(9-1)%20Computer%20Science%20Lesson%202%20Activity%203)’ you can help us to ensure that our resources work for you. When the email template pops up please add additional comments if you wish and then just click ‘Send’. Thank you.

If you do not currently offer this OCR qualification but would like to do so, please complete the Expression of Interest Form which can be found here: [www.ocr.org.uk/expression-of-interest](http://www.ocr.org.uk/expression-of-interest)

**OCR Resources**: *the small print*OCR’s resources are provided to support the teaching of OCR specifications, but in no way constitute an endorsed teaching method that is required by the Board, and the decision to use them lies with the individual teacher. Whilst every effort is made to ensure the accuracy of the content, OCR cannot be held responsible for any errors or omissions within these resources.   
© OCR 2016 - This resource may be freely copied and distributed, as long as the OCR logo and this message remain intact and OCR is acknowledged as the originator of this work.

OCR acknowledges the use of the following content: n/a

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: [resources.feedback@ocr.org.uk](mailto:resources.feedback@ocr.org.uk)