Component 1: Program construction

Program translation

In order to write software a programmer will use a programming language to write code in a way that is understandable to the programmer.

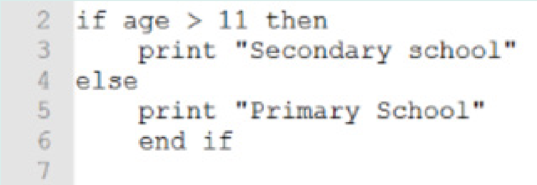
However, the processor will not be able to run the programmer's code and therefore it has to be translated into machine code that can be processed.

Translators are software that converts programs from one language to another. There are three types of translator: compiler, interpreter and assembler.

Term	Definition
Compiler	A compiler is a program that converts high level language programs into machine code for execution at a later time. The entire program is converted at one time.
Interpreter	An interpreter is a program which converts code one line at a time, into machine code and executes it.
Assembler	An assembler is a program which converts the low-level assembly programming language into machine code.

Interpreters and compilers are used to translate high level language programs.

Term	Definition

High level Language	A high level language allows programmers to write programs that are independent of a particular type of computer. Such languages are considered high level because they are closer to spoken language and further away from machine code.
	
	`if age > 11 then`
	`    print "Secondary school"`
	`else`
	`    print "Primary School"`
	`    end if`

Source code	Programming code that has not yet been translated into an executable file.
Object code	Translated code that can be executed by a computer.

Interpreter

An interpreter reads a statement from the high level code and converts that line of the source code into object code and executes it straight away. If there is an error in a line of source code the interpreter will stop the translation process.

Interpreters are often used in the development of a program as they make debugging the code easier.

No executable file is produced, and the program is translated from the beginning each time it is run.

Compiler

A compiler translates the entire source code program into object code before the program is executed producing an executable file.

An advantage of a compiler is that once the code has been translated it can be run many times without having to be translated again.

A disadvantage of a compiler is that if there is an error in the code the translation will carry on and the error is not reported until the end of the process.

The compilation process involves the following steps:

Lexical analysis:

- Comments (annotations) and unneeded spaces are removed.
- Keywords, constants and identifiers are replaced by tokens

Term	Examples
Keywords	IF, WHILE, DO, REPEAT
Constants	100, 3.1429, -73
Identifiers	Variable names, subroutine names

Symbol table construction

- A symbol table is created which holds the addresses of variables, labels and subroutines.