

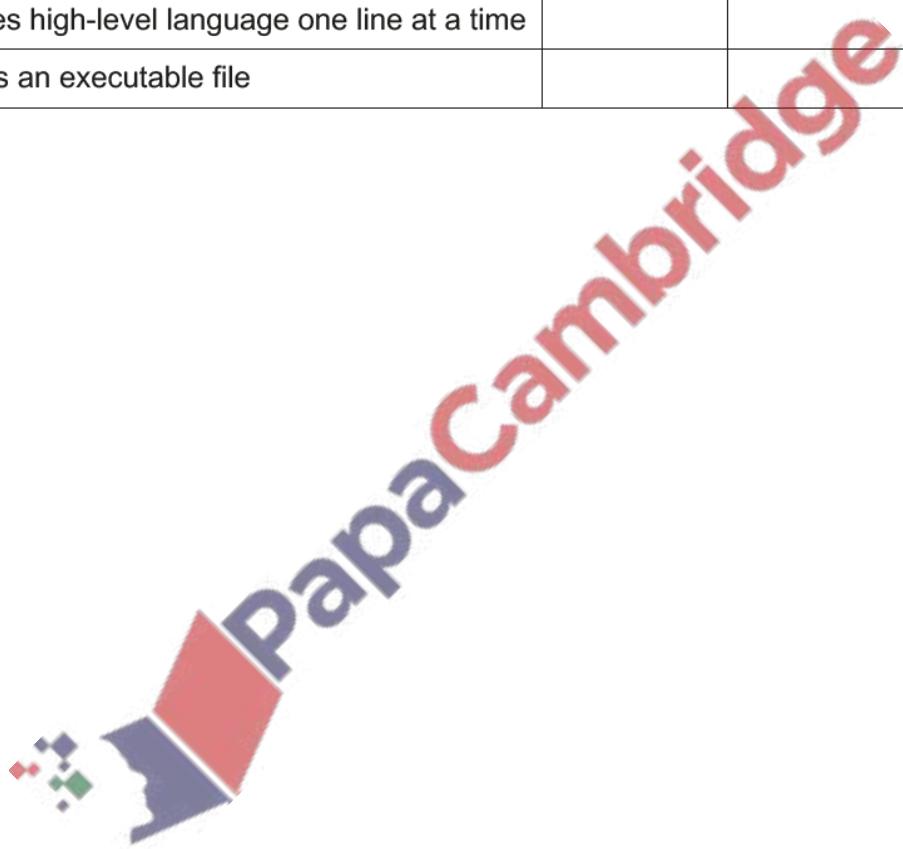
1. Mar/2020/Paper_12/No.4

Assemblers, compilers and interpreters are types of translators.

Tick (✓) to show which statements apply to each translator. Each statement may apply to more than one type of translator.

Statement	Assembler (✓)	Compiler (✓)	Interpreter (✓)
Translates low-level language to machine code			
Translates high-level language to machine code			
Produces error messages			
Translates high-level language one line at a time			
Produces an executable file			

[5]



Programs can be written in a low-level language.

- (a) Identify **three** features of a low-level language.

Feature 1

Feature 2

Feature 3

[3]

- (b) Give **two** examples of a low-level language.

Example 1

Example 2

[2]

- (c) Give **one** drawback of writing programs in a low-level language, instead of a high-level language.

.....

..... [1]

- (d) A low-level language needs to be converted to binary before it can be processed by a computer.

- (i) Give the **8-bit binary** value of the two denary values:

180

201

[2]

Working space

.....

.....

.....

- (ii) Give the **12-bit binary** value of the denary value **250**.

..... [1]

Working space

.....

.....

.....

.....

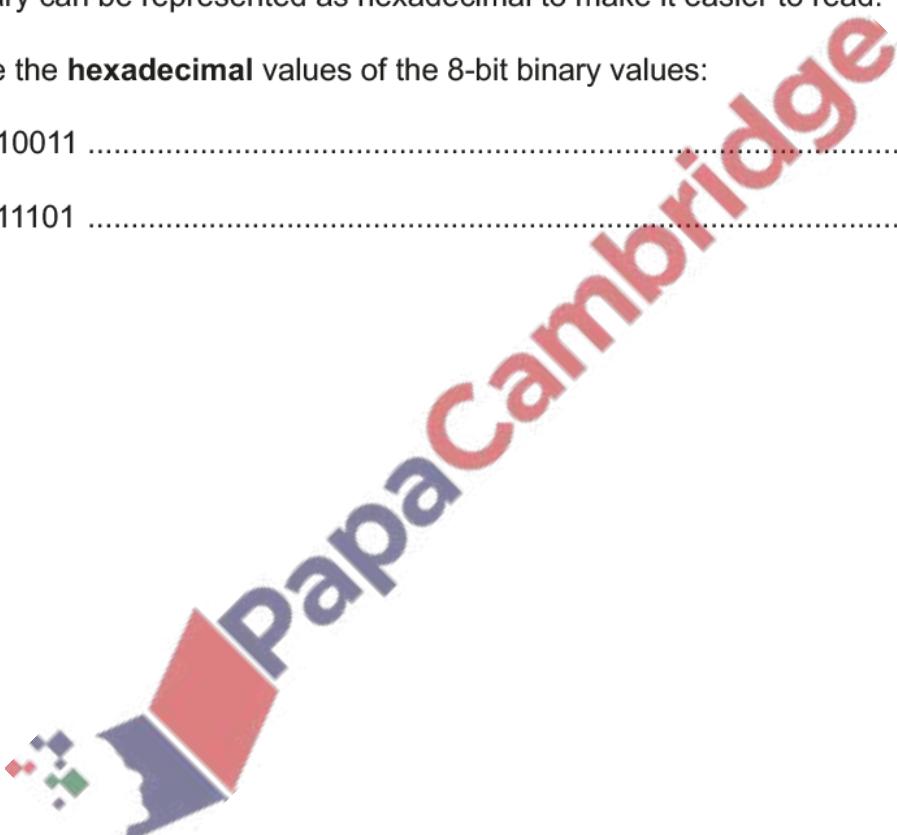
- (iii) Binary can be represented as hexadecimal to make it easier to read.

Give the **hexadecimal** values of the 8-bit binary values:

10010011

00011101

[2]



3. June/2020/Paper_12/No.2

Both an interpreter and a compiler can be used when writing a program in a high-level language.

- (a) Explain why a programmer would make use of both an interpreter **and** a compiler.

.....
.....
.....
.....
.....
.....
.....

[4]

- (b) Give **three** reasons why a programmer would choose to write a program in a high-level language, instead of a low-level language.

Reason 1

.....

Reason 2

.....

Reason 3

.....

[3]

