Name _____ Date _____

4.2 Translating

Getting started

1 Tick (✓) to show whether each statement refers to a high-level or low-level language.

Statement	High-level	Low-level
Uses English-like words.		
It is more difficult to find errors.		
Easier for people to write.		
Machine code is one example.		
Assembly language is one example.		
It needs translating using an interpreter or compiler before it can be run.		

2 Tick (✓) to show whether each statement refers to a compiler, an interpreter, or both.

Statement	Compiler	Interpreter
Converts a high-level language into a low-level language.		
Most appropriately used when the program is finished.		
Converts one statement, then runs it.		
Reports syntax errors in the code.		
Converts all the code before executing it.		
Most appropriately used when writing the program.		
Creates an executable file.		
Allows you to correct an error and then continue running from that position.		

Practice

For each scenario, identify if an interpreter or compiler is most appropriate. Justify your choice.

Scenario	Interpreter or compiler	Justification
A student is writing a program for their homework and needs to test sections of their program whilst writing it.		
A game designer has written a program and wants to distribute it to beta testers.		

Challenge

The following language is written in assembly language:

```
LDD 23
ADD 1
STO 23
LDD 24
SUB 5
ADD 23
OUT
```

The command (operator) is the first 3 characters.
The data or memory location (operand) is the numbers after each command.

The table gives the binary equivalent for the operators.
The operand is translated as a denary number into binary.

Operator	Operand
OUT	10010101
SUB	10010110
ADD	10010111
STO	10011000
LDD	10011001

Perform the function of an assembler by translating the program into binary.