

3(a)	Any two from: • Low-level language • Uses mnemonics • Used to communicate directly with the computer hardware
3(b)	Assembler
3(c)	B

7(a)	• Language: assembly • Translator: assembler
7(b)	Any four from • Each statement is translated and executed before the next • Translator stops when an error is found • ... can correct and continue from where error found // real time debugging when/where error found • ... does not need to retranslate all the code again • Can test sections of the code • Can test without all the program code being functional/accurate/complete
7(c)	Any three from: Example • Code editor • Run-time environment • Error diagnostics // by example • Auto-complete • Auto-correct • Prettyprint // syntax highlighting • Collapse/expand block • Auto-documentation

6(a)	Any one from: • Computer system that is designed to perform dedicated/single function • Computer system that contains a microprocessor (and)/dedicated hardware • Computer system that is built into a larger system
------	---

6(b)	Assembler
6(c)	Any one from: • Convert to hexadecimal • Convert to denary • A character set can be used
6(d)	Any one from: • More control over manipulating the hardware • Faster execution for testing than a high-level language • They can use machine specific instructions
6(e)	Six from (MAX 4 for stating features): Code editor that allows the user to enter and amend code in their program Run-time environment ... that allows a program to be run and see the outputs of their program Error-diagnostic ... to show the programmer where there are errors in the program Auto-completion ... to give the user options/suggestions of key commands to select Auto-correction ... to correct a command that has a minor misspelling Prettyprint ... changes the colour of key commands so they are easy to identify

7	Any four from: • It is a piece of software • ... used to write/develop/edit code • ... used test/debug the code • ... with features such as auto-completion // any suitable example • To translate the code to low level language/machine code
---	--

7(a)	Any two from: • Close to the language processed by computers • May use mnemonics • An example is assembly language/machine code
7(b)	Any two from: • Can directly manipulate the hardware • No requirement for the program to be portable • Program will be more memory efficient • No requirement for a compiler/interpreter • Quicker to execute • Can use specialised hardware

5(a)	• C
5(b)(i)	Any three from: • It translates the (high-level language) to low-level language/object code/machine code • It translates all the code before it is executed • It creates an executable file
5(b)(ii)	Any two from: • It creates an error report after trying to compile • ... displaying all errors in the code • ... that require correction before execution can take place
5(c)	Any three from: e.g. • Code editors • Run-time environment • Built-in interpreter • Error diagnostics • Auto-completion • Auto-correction • Prettyprint