4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

Question	Answer	Marks
9(a)	Any **one** from: – Operating system // Interrupt handler	1
9(b)	Any **five** from: e.g. – Key press generates the interrupt – Interrupt given a priority – Interrupt is sent to CPU – Interrupt is placed in a queue – CPU stops current task to check the queue/service the interrupt … – … using an interrupt service routine – If key press is highest priority the interrupt is processed	5
9(c)(i)	Any **two** suitable hardware example e.g.: – Moving the mouse – Clicking a mouse button – Plugging in a device – Paper jam in printer – Printer out of paper	2
9(c)(ii)	Any **two** suitable software examples e.g.: – Division by zero – Two processes accessing the same memory location – Null value	2

Question	Answer	Marks
1(a)	Any **two** from e.g.: – Touchscreen – Microphone – Keyboard – Keypad – Digital camera – Sensor // by example – Biometric device – Button	2
1(b)	Any **one** from e.g.: – Screen – Speaker – LED/Light – Actuator/Motor	1
1(c)(i)	– 8	1
1(c)(ii)	– 1024	1
1(d)	Any **three** from: – It **performs** the basic functions of a computer – It **manages** the hardware – It provides a platform to run software – It provides a user interface – It performs tasks such as (any example of function of an operating system)	3

Question	Answer	Marks
4	**One** mark for each correct term in the correct place: • System • Application • Operating • Hardware	4

4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

Question	Answer	Marks
7(a)	Any **two** from: • Close to the language processed by computers • May use mnemonics • An example is assembly language/machine code	2
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	2

Question	Answer	Marks
5(a)	• C	1
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	3
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	2

Question	Answer	Marks
5(c)	Any **three** from: e.g. • Code editors • Run-time environment • Built-in interpreter • Error diagnostics • Auto-completion • Auto-correction • Prettyprint	3

Question	Answer	Marks
10(a)	**Two** from: • System software provides services that the computer requires • … whereas application software provides services that the user requires **One** from (system software): • Utility software // by example e.g. defragmentation software, antivirus, firewall • Operating system **One** from (application software): • Any suitable example of an application e.g. word processor, web browser, video-editing software	4
10(b)	• Secondary storage // HDD // SSD	1

Question	Answer	Marks
4(a)	• B	1
4(b)(i)	• Machine code // low-level language // object code	1
4(b)(ii)	• Interpreter	1
4(b)(iii)	• Compiler	1
4(b)(iv)	• Compiler	1

4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

Question	Answer	Marks
11(a)	• Operating system	1
11(b)	Any **one** from: • Create a file • Copy a file • Open a file • Close a file • Move a file • Delete a file • Rename a file • Save a file • Sort files	1
11(c)	Any **two** from: e.g. • Keeping track of the status of each memory location • Managing the movement of data to and from RAM • Checks that processes have enough memory located to them • Makes sure that two processes don't try to access the same memory location • Manage the transfer of pages between virtual memory and RAM • Allows multitasking	2
11(d)	• Interrupt	1

Question	Answer	Marks
2(a)	No mark for choice. Any **four** from matching choice. High-level • Easier for programmer to read/write/understand/edit • … therefore, the programmer is less likely to make mistakes // can write in shorter timeframe • Easier to **debug** // Easier to find/correct **errors** • …so, the programmer can find and correct errors in less time • Game will be machine independent // Game will be portable (between hardware) • …the game can be used on any computer without a need for understanding of the hardware / compilation for that hardware • Programmer can focus on the problem instead of the manipulation of memory/hardware Low-level • More **memory/RAM** efficient • … 3D graphics will have high memory consumption anyway • Allows direct manipulation of memory • … allows for more efficient control/response time • Allows for use of specialised hardware	4
2(b)	**Two** from for each compiler and interpreter. Compiler • Checks all code **before** executing any code • Produces error report with all errors found for the whole code (before translating/running any of the code) • Produces executable file Interpreter • Checks/translates one line of code **and then** executes it before moving on to the next line • Stops when an error is found • … when corrected the program can be run from the same position // allows error correction in real time	4

4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

Question	Answer	Marks
7(a)	• Low-level language	1
7(b)	• Assembler	1
7(c)	Any **two** from: • He can directly access the hardware • He can use special machine-dependent instructions • There is no need for the program to be portable • Smaller file size // takes up less storage space • More efficient use of memory • Programs will be more time efficient when running	2
7(d)	Any **two** from: • Programs are not portable • It is complex to learn • Difficult to debug	2

Question	Answer	Marks
6(a)(i)	• They both report/check for errors	1
6(a)(ii)	**Four** from (**MAX 2** per translator): • An interpreter translates the code line by line (and executes each line immediately) • … whereas a compiler translates the whole code **at the same time** (before executing it) • A compiler produces an executable file • … but an interpreter does not • An interpreter is required to run the code each time if used • … whereas a compiler is not • An interpreter stops and reports an error as it is encountered • … whereas a compiler creates a report of all errors at the end of translation • An interpreter will run code up to the point it finds an error • … whereas a compiler will not run the code at all if an error is found	4

Question	Answer	Marks
4(a)	Any **three** from: • It uses English-like statements • It needs to be converted to **machine code** (to be processed by a computer) • … using a translator • It is portable • One line of code can perform multiple commands	3

Question	Answer	Marks
8	**One** mark per each correct term in the correct place • high-level language • line by line • all at once • executable file • is not required • debugging	6

4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

Question	Answer	Marks
7(a)(i)	1 mark for when e.g. • Development // when writing the program // when debugging 1 mark for explanation to max 2 from: e.g. • … easier to debug • …stops when an error is detected • …reports one error at a time • …can correct errors in run-time // correct the line and then continue running from that point • …can test one section without the rest of the code being completed	3
7(a)(ii)	1 mark for when e.g. • After completion // For distribution // For final/repeated testing 1 mark each to max 2 from: e.g. After completion • It creates an executable file • …than can be distributed without source code • …so that other people cannot edit/view the code • …so end users do not need translator software // so end users do not need to compile/interpret each time • …so it is machine/platform independent (usually) In final testing • It creates an executable file • …do not need to retranslate for each test sequence • …can test repeatedly with different data faster	3

Question	Answer	Marks
8(a)	– High-level	1
8(b)(i)	**One** mark for the correct translator, **two** marks for the benefit(s). – Interpreter – Easier to debug – … as errors are immediately reported when detected – Compiler – All errors are reported in a single report – … meaning they can all be fixed at the same time – No need to recompile code every time a test is run	3
8(b)(ii)	**One** mark for the correct translator, **two** marks for the benefits. – Compiler – Creates an executable file – … so, translator is no longer needed to run it – Source code cannot be stolen // can be provided without the source code	3

4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5</td><td>One mark per each correct row

<table>
<tr><th>Statement</th><th>High-level language (✓)</th><th>Assembly language (✓)</th><th>Machine code (✓)</th></tr>
<tr><td>It requires a translator to be processed by a computer</td><td>✓</td><td>✓</td><td></td></tr>
<tr><td>It is an example of low-level language</td><td></td><td>✓</td><td>✓</td></tr>
<tr><td>It uses mnemonics</td><td></td><td>✓</td><td></td></tr>
<tr><td>It uses English-like statements</td><td>✓</td><td></td><td></td></tr>
<tr><td>It can be used to directly manipulate hardware in the computer</td><td></td><td>✓</td><td>✓</td></tr>
<tr><td>It is portable</td><td>✓</td><td></td><td></td></tr>
</table></td><td>6</td></tr>
</table>

Question	Answer	Marks
6	**One** from: • Interrupt Any **two** from e.g.: • Paper jam • Paper tray empty • Any change of task example • Any error occurrence example	3

Question	Answer	Marks
3	**One** mark for each correct term: **Term** / **Description** hardware — A collective term for the physical components of the computer system. application software — A type of software that provides services that the user requires and allows the user to perform tasks on the computer. operating system — A type of software that manages the main functions of the computer, including managing files and managing memory. firmware — A type of software that is stored in the read only memory (ROM). It includes the basic input output system (BIOS) and the bootloader.	4

4.1 & 4.2 – Types of Software, Interrupts. IDEs & Translators
ANSWERS

1 (a) Any **two** from:

- direct access to computer processor / special hardware // machine dependent instructions
- uses up less memory
- can increase the speed of processing a program // executes instructions faster

[2]

(b)

Statements	Interpreter (✓)	Compiler (✓)
Translates the source code into machine code all at once		✓
Produces an executable file in machine code		✓
Executes a high-level language program one instruction at a time	✓	
Once translated, the translator does not need to be present for the program to run		✓
An executable file is produced		✓

[5]