4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

4 - Software

4.1 Types of software and interrupts				
1	Describe the difference between system software and application software and provide examples of each			
2	Describe the role and basic functions of an operating system			
3	Understand how hardware, firmware and an operating system are required to run applications software			
4	Describe the role and operation of interrupts			

More Guidance:

4.1 Types of software and interrupts

Candidates should be able to:	Notes and guidance
1 Describe the difference between system software and application software and provide examples of each	• System software provides the services that the computer requires, including operating system and utility software • Application software provides the services that the user requires
2 Describe the role and basic functions of an operating system	• Including: – managing files – handling interrupts – providing an interface – managing peripherals and drivers – managing memory – managing multitasking – providing a platform for running applications – providing system security – managing user accounts
3 Understand how hardware, firmware and an operating system are required to run applications software	• Applications are run on the operating system • The operating system is run on the firmware • The bootloader (firmware) is run on the hardware
4 Describe the role and operation of interrupts	• Including: – how an interrupt is generated – how it is handled using an interrupt service routine – what happens as a result of the interrupts • Software interrupts include division by zero and two processes trying to access the same memory location • Hardware interrupts include pressing a key on the keyboard and moving the mouse

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

4.2 Types of programming language, translators and integrated development environments (IDEs)				
1	Explain what is meant by a high-level languageand a low-level language, including the advantages and disadvantages of each			
2	Understand that assembly language is a form of low-level language that uses mnemonics, and that an assembler is needed to translate an assembly language program into machine code			
3	Describe the operation of a compiler and an interpreter, including how high-level language is translated by each and how errors are reported			
4	Explain the advantages and disadvantages of a compiler and an interpreter			
5	Explain the role of an IDE in writing program code and the common functions IDEs provide			

More Guidance:

4.2 Types of programming language, translators and integrated development environments (IDEs)

Candidates should be able to:	Notes and guidance
1 Explain what is meant by a high-level language and a low-level language, including the advantages and disadvantages of each	• Advantages and disadvantages include: – ease of reading and writing code, e.g. low-level is hard to read – ease of debugging code – machine independence – direct manipulation of hardware
2 Understand that assembly language is a form of low-level language that uses mnemonics, and that an assembler is needed to translate an assembly language program into machine code	
3 Describe the operation of a compiler and an interpreter, including how high-level language is translated by each and how errors are reported	• A compiler translates the whole code at once before executing it, producing an executable file • An interpreter translates and executes the code line-by-line • A compiler provides an error report for the whole code if errors are detected • An interpreter stops execution when an error is found
4 Explain the advantages and disadvantages of a compiler and an interpreter	• To include an understanding that an interpreter is mostly used when developing a program and a compiler is used to translate the final program
5 Explain the role of an IDE in writing program code and the common functions IDEs provide	• Including: – code editors – run-time environment – translators – error diagnostics – auto-completion – auto-correction – prettyprint

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

9 An interrupt is a type of signal that is used in a computer.

(a) State the name of the type of software that manages interrupts.

..... [1]

(b) Describe how interrupts are used when a key is pressed on a keyboard.

.....

.....

.....

.....

.....

.....

.....

.....

.....

..... [5]

(c) Interrupts can be hardware based or software based.

A key press is one example of a hardware interrupt.

(i) Give **two** other examples of a hardware interrupt.

1 .....

.....

2 .....

..... [2]

(ii) Give **two** examples of a software interrupt.

1 .....

.....

2 .....

..... [2]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

1 A mobile telephone has built-in input and output devices.

(a) Give **two** examples of an input device that would be built into a mobile telephone.

1 .....

2 .....
[2]

(b) Give **one** example of an output device that would be built into a mobile telephone.

..... [1]

(c) The data storage in the mobile telephone can be measured using different units of measurement.

(i) State how many bits are equal to a byte.

..... [1]

(ii) State how many kibibytes (KiB) equal a mebibyte (MiB).

..... [1]

(d) The mobile telephone has an operating system.

Describe the purpose of the operating system.

.....

.....

.....

.....

.....

..... [3]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

4 Complete the statements about different types of software.

Use the terms from the list.

Some of the terms in the list will **not** be used. You should only use a term once.

application assembly language bootloader central processing unit (CPU)
firmware hardware operating output system user

..... software provides the services that the computer requires; an example is utility software.

..... software is run on the operating system.

The ..... system is run on the firmware, which is run on the ..... .

[4]

7 A programmer uses a low-level language to write a computer program for a vending machine.

(a) Describe what is meant by a low-level language.

.....

.....

.....

..... [2]

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

1 .....

.....

2 .....

.....

[2]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

5 A programmer writes a computer program using a high-level language.

(a) Tick (✓) **one** box to show which statement is correct about writing computer programs in a high-level language.

A Mnemonics are used to create instructions. ☐

B The computer program is harder to debug than a low-level language program. ☐

C The computer program is machine independent. ☐

D The hardware of the computer can be directly manipulated. ☐

[1]

(b) The programmer uses a compiler to translate the computer program.

(i) Describe how the compiler translates the computer program.

.....

.....

.....

.....

.....

..... [3]

(ii) Describe how the compiler reports errors.

.....

.....

.....

..... [2]

(c) The programmer uses an integrated development environment (IDE) to create the computer program.

One function of the IDE is that it has the built-in compiler.

Give **three** other common functions of an IDE.

1 .....

2 .....

3 .....

[3]

10 A user has both system software and application software installed on their computer.

(a) Describe the difference between system software and application software.

Give an example of each software in your answer.

.....

.....

.....

.....

.....

.....

..... [4]

(b) State which component in the computer would store both types of software when the power is turned off.

..... [1]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

4 A programmer writes a computer program in a high-level language.

(a) Tick (✓) **one** box to show which statement is a benefit of writing a program in a high-level language, instead of a low-level language.

A The program can directly manipulate the hardware. ☐

B The program is machine independent. ☐

C The program is more memory efficient. ☐

D The program is quicker to execute. ☐

[1]

(b) Translators are used to translate the high-level language so that it can be processed by the computer.

(i) State what the high-level language is translated into.

..... [1]

(ii) One translator converts and executes the code line by line.

Identify which type of translator would do this.

..... [1]

(iii) One translator creates an error report displaying all the errors in the code before it can be executed.

Identify which type of translator would do this.

..... [1]

(iv) One translator creates an executable file.

Identify which type of translator would do this.

..... [1]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

11 Software is installed on a computer to manage files, memory and multitasking.

(a) State the name of the software that can do these tasks.

..... [1]

(b) Give **one** task that the software allows the user to do to manage files.

..... [1]

(c) Describe what is meant by managing memory.

.....

.....

.....

..... [2]

(d) A signal is sent within the computer to allow multitasking to occur.

State the name of this type of signal.

..... [1]

2 A programmer has designed a three-dimensional (3D) interactive computer game. They are going to develop a program for the game. The program needs to run efficiently, but it must also be developed as soon as possible.

(a) Tick (✓) **one** box to identify whether the programmer should use a high-level language or a low-level language to develop the program.

Explain the reasons for your choice.

High-level language ☐

Low-level language ☐

Reasons for your choice .....

.....

.....

.....

.....

.....

.....

..... [4]

(b) If the programmer chooses a high-level language, they can use a compiler or an interpreter to translate the high-level language into a low-level language.

Describe the operation of a compiler and of an interpreter.

Compiler .....

.....

.....

.....

Interpreter .....

.....

.....

.....

[4]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

7 Cameron writes software for coffee machines. He uses assembly language to write his software.

(a) Tick (✓) to show whether assembly language is an example of a high-level language, a low-level language or machine code.

	Tick (✓)
High-level language	☐
Low-level language	☐
Machine code	☐

[1]

(b) Identify the translator that is required for assembly language.

..... [1]

(c) Give **two** reasons why Cameron chooses to write the software for the coffee machines in assembly language.

Reason 1 .....

.....

Reason 2 .....

.....

[2]

(d) Give **two** drawbacks of using assembly language to write programs.

Drawback 1 .....

.....

Drawback 2 .....

.....

[2]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

6 A compiler and an interpreter are two different types of translator.

(a) One similarity between a compiler and an interpreter is that they both translate high-level language into machine code.

(i) Give **one** other similarity between a compiler and an interpreter.

.....

..... [1]

(ii) Explain **two** differences between a compiler and an interpreter.

.....

.....

.....

.....

.....

.....

.....

..... [4]

4 Jason is a programmer who writes computer programs in a high-level language.

(a) Describe what is meant by a high-level language.

.....

.....

.....

.....

.....

..... [3]

8 The paragraph provides information about translators.

Complete the paragraph using the most appropriate terms from the list. **Not** all terms in the list need to be used.

- all at once
- assembler
- assembly language
- binary
- commenting
- compiling
- debugging
- executable file
- high-level language
- interpreted
- is **not** required
- is required
- line by line
- low-level language

A compiler and an interpreter translate ..... into machine code. An interpreter does this by translating and executing the code ..... and a compiler does this by translating and executing the code ..... .

One benefit of a compiler is that it creates an ..... , which means that a translator ..... each time to run the file.

..... is easier to do with an interpreter because it stops translating and reports an error at the place where it occurs in the code.

[6]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

7 Ishani is a software developer who is creating a new computer game.

(a) Ishani uses an interpreter and a compiler at different stages of the game creation.

(i) Explain when it is most appropriate for Ishani to use an interpreter.

.....

.....

.....

.....

.....

..... [3]

(ii) Explain when it is most appropriate for Ishani to use a compiler.

.....

.....

.....

.....

.....

..... [3]

8 Victoria develops a computer game to sell on a gaming website. She writes her program using English-like statements.

(a) State which type of programming language Victoria is using.

..... [1]

(b) Victoria uses **two** different types of translator when creating the program for the computer game.

State which translator is the most suitable for the given tasks.

Give the benefits of using that translator for the task.

You must choose a different translator for each task.

(i) To translate the code during development of the game.

Translator .....

Benefits .....

.....

.....

.....

.....

.....

[3]

(ii) To translate the final program and upload to the website for distribution, without the source code.

Translator .....

Benefits .....

.....

.....

.....

.....

.....

[3]

5 Six statements are given about programming languages.

Tick (✓) to show whether each statement applies to high-level language, assembly language or machine code. Some statements may apply to more than one type of programming language.

Statement	High-level language (✓)	Assembly language (✓)	Machine code (✓)
it requires a translator to be processed by a computer			
it is an example of low-level language			
it uses mnemonics			
it uses English-like statements			
it can be used to directly manipulate hardware in the computer			
it is portable			

[6]

6 A computer sends the data for an image to the print buffer, to wait to be processed for printing, until the buffer is full.

When the buffer is no longer full, it sends a signal back to the computer to state it is ready for more data.

State the name of this signal and give **two** other examples of when this type of signal is used.

Signal name .....

Examples .....

.....

.....

.....

.....

[3]

4.1 & 4.2 – Types of Software, Interrupts, IDEs & Translators
QUESTIONS

3 The table contains **four** descriptions about a computer system.

Complete the table by writing the correct term for each description.

Term	Description
.....	A collective term for the physical components of the computer system.
.....	A type of software that provides services that the user requires and allows the user to perform tasks on the computer.
.....	A type of software that manages the main functions of the computer, including managing files and managing memory.
.....	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
QUESTIONS

1 **(a)** Give **two** reasons why a programmer would choose to write code in a low-level language.

1 .....

.....

.....

2 .....

.....

.....

[2]

(b) High-level languages require either an interpreter or a compiler to translate the program.

The table below lists a number of statements about language translators.

Tick (✓) to show which statements refer to interpreters and which refer to compilers.

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]