Learning Programme

Fundamentals of computer systems – AS Level

Topic/Content	Objectives/Skills	Homework	Assessment	Stretch & Challenge (Thirst for Learning)
Hardware and Software	• Understand the relationship between hardware and software and be able to define the terms: ○ hardware ○ software • Explain what is meant by: ○ system software ○ application software. • Understand the need for, and attributes of, different types of software • Understand the need for, and functions of the following system software: ○ operating systems (OSs) ○ utility programs ○ libraries ○ translators (compiler, assembler, interpreter).		Worksheets Q and A’s Homework worksheets Little Man Computer task booklet End of unit test	Look further into the Little Man Computer Use logic gate website to create the circuits and see how they would work in real life situations
Role of an operating system (OS)	• Understand that a role of the operating system is to hide the complexities of the hardware. • Know that the OS handles resource management, managing hardware to allocate processors, memories and I/O devices among competing processes.	Operating System presentation		
Classification of programming languages	• Show awareness of the development of types of programming languages and			

	their classification into low-and high-level languages. • Know that low-level languages are considered to be: ○ machine-code ○ assembly language. • Know that high-level languages include imperative high level-language. • Describe machine-code language and assembly language. • Understand the advantages and disadvantages of machine-code and assembly language programming compared with high-level language programming. • Explain the term 'imperative high-level language' and its relationship to low-level languages.			
Types of program translator	• Understand the role of each of the following: ○ Assembler ○ Compiler ○ Interpreter. • Explain the differences between compilation and interpretation. Describe situations in which each would be appropriate. • Explain why an intermediate language such as bytecode is produced as the final output by some compilers and how it is subsequently used. • Understand the difference between source and object (executable) code.	Past exam style questions		

Logic gates	• Construct truth tables for the following logic gates: ○ NOT ○ AND ○ OR ○ XOR ○ NAND ○ NOR. • Be familiar with drawing and interpreting logic gate circuit diagrams involving one or more of the above gates. • Complete a truth table for a given logic gate circuit. • Write a Boolean expression for a given logic gate circuit. • Draw an equivalent logic gate circuit for a given Boolean expression.	Watch the Craig and Dave video on Boolean Algebra		
Boolean algebra	• Be familiar with the use of Boolean identities and De Morgan’s laws to manipulate and simplify Boolean expressions.	Complete the Boolean algebra worksheet		