Year one

Week 1

Lesson one	Lesson two	Lesson three	Lesson four
Introduction to the chosen programming language and its development environment. Application of concepts, operations and simple data constructs to complete simple programs and develop the confidence to experiment.			Practical experience of programming.

Week 2

Lesson one	Lesson two	Lesson three	Lesson four
Introduction to the chosen programming language and its development environment. Application of concepts, operations and simple data constructs to complete simple programs and develop the confidence to experiment.			Practical experience of programming.

Week 3

Lesson one	Lesson two	Lesson three	Lesson four
Introduction to the chosen programming language and its development environment. Application of concepts, operations and simple data constructs to complete simple programs and develop the confidence to experiment.			Practical experience of programming.

Week 4

Lesson one	Lesson two	Lesson three	Lesson four
Progression to a wider range of program tasks, including 1- and 2-dimensional arrays.			Practical experience of programming.

Week 5

Lesson one	Lesson two	Lesson three	Lesson four
Number systems.	Binary and other representations and discuss their uses, advantages and limitations.		Summative assessment.

Week 6

Lesson one	Lesson two	Lesson three	Lesson four
Coding systems: introduction.	Binary and other representations and discuss their uses, advantages and limitations.		Practical: info coding.

Week 7

Lesson one	Lesson two	Lesson three	Lesson four
Describe and explain the use of parity bits, majority voting and check digits.			Images and sound: Intro.

Week 8

Lesson one	Lesson two	Lesson three	Lesson four
Representation of images.			Digital photography.

Week 9

Lesson one	Lesson two	Lesson three	Lesson four
Representation of sound.			Practical: eg MIDI.

Week 10

Lesson one	Lesson two	Lesson three	Lesson four
Compression theory.	Compression in practice.	Encryption theory.	Summative assessment.

Week 11

Lesson one	Lesson two	Lesson three	Lesson four
Experience in working with ciphers and computerised encryption.			Abstraction and automation.

Week 12

Lesson one	Lesson two	Lesson three	Lesson four
Abstraction and automation: understand the concepts and gain experience in the techniques and processes.			

Week 13

Lesson one	Lesson two	Lesson three	Lesson four
Finite state machines: Manipulate state transition diagrams and state transition tables for FSMs.			

Week 14

Lesson one	Lesson two	Lesson three	Lesson four
Communication and networking part 1.			

Week 15

Lesson one	Lesson two	Lesson three	Lesson four
Communication and networking part 2.			

Week 16

Lesson one	Lesson two	Lesson three	Lesson four
Gain experience in solving problems by iterating stages of development, including: Abstracting, Structuring, Coding, Testing – a guided, small-scale project.			

Week 17

Lesson one	Lesson two	Lesson three	Lesson four
Gain experience in solving problems by iterating stages of development, including: Abstracting, Structuring, Coding, Testing – a guided, small-scale project.			

Week 18

Lesson one	Lesson two	Lesson three	Lesson four
Project finalisation and evaluation.		Communication and networking part 3.	

Week 19

Lesson one	Lesson two	Lesson three	Lesson four
Consequences of uses: individual (moral), social (ethical), legal and cultural issues and opportunities.			

Week 20

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of computer systems. This includes hardware, software, programming languages and program translators.			

Week 21

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of computer systems.			Logic gates.

Week 22

Lesson one	Lesson two	Lesson three	Lesson four
Logic gates and Boolean algebra.			Summative assessment.

Week 23

Lesson one	Lesson two	Lesson three	Lesson four
Internal hardware components.			

Week 24

Lesson one	Lesson two	Lesson three	Lesson four
Stored program concept: the structure and role of the processor and its components.			

Week 25

Lesson one	Lesson two	Lesson three	Lesson four
Stored program concept, including the Fetch-Execute cycle.			

Week 26

Lesson one	Lesson two	Lesson three	Lesson four
External hardware devices: main characteristics, purposes and suitability of input, output and secondary storage devices.			

Week 27

Lesson one	Lesson two	Lesson three	Lesson four
Revision and preparation for internal examination and/or	Experience of software development cycle and/or	Research, definition and planning for the assessed project and/or	Visits to or from practitioners and professionals.

Week 28

Lesson one	Lesson two	Lesson three	Lesson four
Revision and preparation for internal examination and/or	Experience of software development cycle and/or	Research, definition and planning for the assessed project and/or	Visits to or from practitioners and professionals.

Week 29

Lesson one	Lesson two	Lesson three	Lesson four
Revision and preparation for internal examination and/or	Experience of software development cycle and/or	Research, definition and planning for the assessed project and/or	Visits to or from practitioners and professionals.

Week 30

Lesson one	Lesson two	Lesson three	Lesson four
Revision and preparation for internal examination and/or	Experience of software development cycle and/or	Research, definition and planning for the assessed project and/or	Visits to or from practitioners and professionals.

Year Two

Week 1

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of databases: Model, describe, explain and normalise relational databases; Use SQL with multiple tables. Discuss concurrent access in a client-server database.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 2

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of databases: Model, describe, explain and normalise relational databases; Use SQL with multiple tables. Discuss concurrent access in a client-server database.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 3

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of databases: Model, describe, explain and normalise relational databases; Use SQL with multiple tables. Discuss concurrent access in a client-server database.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 4

Lesson one	Lesson two	Lesson three	Lesson four
Abstract data types. Required types are: Queue, stack, list, graph, tree, hash table, dictionary, vector.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 5

Lesson one	Lesson two	Lesson three	Lesson four
Abstract data types. Required types are:		Software development – revision.	

Lesson one	Lesson two	Lesson three	Lesson four
Queue, stack, list, graph, tree, hash table, dictionary, vector.		The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 6

Lesson one	Lesson two	Lesson three	Lesson four
Abstract data types. Required types are: Queue, stack, list, graph, tree, hash table, dictionary, vector.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 7

Lesson one	Lesson two	Lesson three	Lesson four
Paper 1 skeleton program.	Summative assessment.	Software development – revision. The computing practical project.	

Lesson one	Lesson two	Lesson three	Lesson four
		The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 8

Lesson one	Lesson two	Lesson three	Lesson four
Numbers with a fractional part, including revision from Year 1.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 9

Lesson one	Lesson two	Lesson three	Lesson four
Floating point – errors etc.	Checksums.	Software development – revision. The computing practical project.	

Lesson one	Lesson two	Lesson three	Lesson four
		The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 10

Lesson one	Lesson two	Lesson three	Lesson four
Stack frames and recursive programming techniques.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 11

Lesson one	Lesson two	Lesson three	Lesson four
Interrupts.	Analogue/digital conversion.	Software development – revision. The computing practical project.	

Lesson one	Lesson two	Lesson three	Lesson four
		The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 12

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of algorithms: Trace and describe a range of algorithms and analyse their time complexity.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 13

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of algorithms: Trace and describe a range of algorithms and analyse their time complexity.		Software development – revision. The computing practical project.	

Lesson one	Lesson two	Lesson three	Lesson four
		The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 14

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of algorithms: Trace and describe a range of algorithms and analyse their time complexity.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 15

Lesson one	Lesson two	Lesson three	Lesson four
Fundamentals of algorithms: Trace and describe a range of algorithms and analyse their time complexity.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of	

Lesson one	Lesson two	Lesson three	Lesson four
		solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 16

Lesson one	Lesson two	Lesson three	Lesson four
Regular expressions and language.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 17

Lesson one	Lesson two	Lesson three	Lesson four
Backus-Naur form.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of	

Lesson one	Lesson two	Lesson three	Lesson four
		solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 18

Lesson one	Lesson two	Lesson three	Lesson four
Vector graphics.	Summative assessment.	Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 19

Lesson one	Lesson two	Lesson three	Lesson four
Programming paradigms: Explain and justify the use of both procedural and object-oriented programming paradigms.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of	

Lesson one	Lesson two	Lesson three	Lesson four
		solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 20

Lesson one	Lesson two	Lesson three	Lesson four
Programming paradigms: Explain and justify the use of both procedural and object-oriented programming paradigms.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 21

Lesson one	Lesson two	Lesson three	Lesson four
Programming paradigms: Explain and justify the use of both procedural and object-oriented programming paradigms.		Software development – revision. The computing practical project. The project allows students to develop their practical skills in the context of	

Lesson one	Lesson two	Lesson three	Lesson four
		solving a realistic problem or carrying out an investigation. Work on development is likely to continue in parallel with study of other topics.	

Week 22

Lesson one	Lesson two	Lesson three	Lesson four
Classification of algorithms.	Finite state machines, including revision from Year 1.		Turing Machines.

Week 23

Lesson one	Lesson two	Lesson three	Lesson four
Turing Machines.			The Halting Problem.

Week 24

Lesson one	Lesson two	Lesson three	Lesson four
The Internet and TCP/IP.			

Week 25

Lesson one	Lesson two	Lesson three	Lesson four
Big Data.			

Week 26

Lesson one	Lesson two	Lesson three	Lesson four
Functional programming.		Paper 1 skeleton program continued.	

Week 27

Lesson one	Lesson two	Lesson three	Lesson four
Exam preparation.			

Week 28

Lesson one	Lesson two	Lesson three	Lesson four
Exam preparation.			

Week 29

Lesson one	Lesson two	Lesson three	Lesson four
Exam preparation.			

Week 30

Lesson one	Lesson two	Lesson three	Lesson four
Exam preparation.			