



YEAR 11 GCSE COMPUTER SCIENCE SUMMER TERM 3 – PAPER 2

'An ambitious curriculum that meets the needs of all'

Medium Term Planning – Programming

Curriculum Intent	Pupils will be taught the following National Curriculum guidelines this term:
Skills/Assessment Objective Links	<p>At the end of this Learning Aim all students should be able to:</p> <ul style="list-style-type: none">• Understand and use data types: integer, real, Boolean, character and string• Declare and use constants and variables• Use input, output and assignment statements• Use random number generation• Write algorithms in pseudocode involving sequences <p>Most students will be able to:</p> <ul style="list-style-type: none">• Use arithmetic operators including MOD and DIV• Use string handling and conversion functions• Use selection and nested selection statements• Use NOT, AND and OR when creating Boolean expressions• Understand and use iteration in an algorithm• Write algorithms in pseudocode involving sequence, selection and iteration• Use one- and two-dimensional arrays in the design of solutions to simple problems• Understand the concept of subroutines• Understand and use basic file handling operations:<ul style="list-style-type: none">– open– read– write– close• Use SQL (Structured Query Language) statements to search for data:<ul style="list-style-type: none">– Formulate criteria involving AND, OR and LIKE– Use SELECT, FROM, WHERE, ORDER BY statements– Use the wildcard * <p>Some students will be able to:</p> <ul style="list-style-type: none">• Learn how to write simple procedures and functions• Understand and use parameters to pass data to procedures and functions• Know that subroutines may use local variables which are accessible only within the subroutine• Use local variables and explain why it is good practice to do so• Explain the advantages of using subroutines in programs• Read from and write to a text file of this Unit all students should be able to:
Numeracy	Arithmetic, BIDMAS, Comparison operators, AND, OR, NOT, ==, !=, <, <=, >, >=, +, -, *, /, MOD, DIV, ^, exponentiation, data types, integer, real,
Literacy	<p>Vocabulary Tier 3: Variables, constants, operators, inputs, outputs, assignment, sequence, selection, iteration, arithmetic operators, Boolean operators, AND, OR, NOT, ==, !=, <, <=, >, >=, +, -, *, /, MOD, DIV, ^, exponentiation, data types, integer, real, Boolean, character, string, casting, string manipulation, file handling, open, read, write, close, records, SQL, arrays, one-dimensional array, two-dimensional array, sub program/subroutine, functions, procedures, random numbers, concatenation, slicing, SQL, SELECT, FROM, WHERE.</p> <p>Vocabulary Tier 2: Arithmetic, open, read, write, close</p> <p>Reading: Worksheets, presentations, answer sheets, exam questions, mark scheme, further reading for homework</p> <p>Writing: Answer on the worksheet via word</p> <p>Oracy: listening and using tier 3 words</p>

Becoming future ready	Careers/Employability: <ul style="list-style-type: none"> Software Architect. Data Scientist. Machine Learning Engineer. Blockchain Developer Cybersecurity Engineer. Cloud Solutions Architect. AI Research Scientist. Full-Stack Developer.
Adaptation	Throughout this topic, quality first teaching will provide differentiation:
QFT/SEND Provision	<p>By product: Learners are asked to present outcomes writing different code, not all code will be equal in style and sophistication, all code will work with teachers input, top end programmers will be set challenges on how to extend code and be expected to conduct a level of independent research</p> <p>By resource: Worksheets are well presented and accessible. Instructions are clearly outlined and separate from the information so that pupils know where to begin and end. Handouts are differentiated by outcome. Resources used will appeal to the range of preferred learning styles of pupils e.g. visual, auditory or kinesthetic learners. Scaffolding of tasks – word frames.</p> <p>By Intervention: By providing different levels of supervision and support depending on coding ability</p> <p>By Progressive Questioning: Exploring pupils' understanding of programming by setting adaptive challenges</p> <p>By Grouping: According to coding ability, prior attainment, gender, social preference, preferred learning style.</p> <p>By Task: Pupils identify targets which are meaningful via level of coding ability</p> <p>By Offering Optional Activities: In class or as homework, to extend learning.</p> <p>This QFT/SEND provision will be explicit within the lesson by lesson schemes of work.</p>
Implementation Curriculum Delivery	To be able to:
Learning Outcomes (Knowledge)	<p>Topic 1 Programming fundamentals Understand and use data types: integer, real, Boolean, character and string Declare and use constants and variables Use input, output and assignment statements Use arithmetic operators including MOD and DIV Use string handling and conversion functions</p> <p>Topic 2 Sequence and selection Use selection and nested selection statements Use NOT, AND and OR when creating Boolean expressions Use random number generation</p> <p>Topic 3 Iteration Understand and use iteration in an algorithm Write algorithms in pseudocode involving sequence, selection and iteration</p> <p>Topic 4 Procedures and Functions Understand the concept of subroutines Learn how to write simple procedures and functions Understand and use parameters to pass data to procedures and functions Know that subroutines may use local variables which are accessible only within the subroutine Use local variables and explain why it is good practice to do so Explain the advantages of using subroutines in programs</p> <p>Topic 6 Records and files Understand and use basic file handling operations: – open</p>



- read
 - write
 - close
- Read from and write to a text file

Topic 7 Introduction to SQL

Use SQL (Structured Query Language) statements to search for data:

- Formulate criteria involving AND, OR and LIKE
- Use SELECT, FROM, WHERE, ORDER BY statements
- Use the wildcard *

Programing end of unit assessment

Current learning to be developed in the future within:

Writing programs for exam questions. Using SQL.

Assessment

See assessment maps for formative and summative assessment opportunities.

Impact

Review assessment results and target pupils that require further support via:-

- Learning conversation
- Changing seating plan
- Plan lessons to address areas of concern in assessment
- Targeted homework based on low performance areas identified in the assessment and marked pieces
- Stretch and challenge high ability pupils by identifying ambitious next steps to expand knowledge

Create a feedback sheet for each student

Each student identifies areas of Green, Amber and Red using Mark Assessment on their feedback sheet

Complete NOW task on areas identified as Amber and Red