PCHS Curriculum Information - Computer Science

Course Title: Computer Science	**Exam Board: OCR**	**Specification Code:J277**

How will students be assessed?
Unit 1: Computer Systems
Written Exam, 50%
Unit 2: Computational thinking, algorithms and programming
Written Exam, 50%

KEY CONTENT

Half Term 1

Introduction to Computer Science GCSE, Careers and Opportunities
2.1.1 Computational thinking - Principles of computational thinking: Abstraction, Decomposition, Algorithmic thinking
1.2.4 Binary - Understanding of base 2 and how computers use binary
2.4.1 Boolean logic - Simple logic diagrams using the operators AND, OR and NOT - Truth tables - Combining Boolean operators using AND, OR and NOT - Applying logical operators in truth tables to solve problems
2.2.2 Data types - The use of data types: Integer, Real, Boolean, Character and string, Casting

Half Term 2

2.2.1 Programming fundamentals - The use of variables, constants, operators, inputs, outputs and assignments - The use of the three basic programming constructs used to control the flow of a program: Sequence, Selection
Iteration (count- and condition-controlled loops) - The common arithmetic operators - The common Boolean operators AND, OR and NOT
2.2.3 Additional programming techniques - The use of basic string manipulation - The use of basic file handling operations: Open, Read, Write, Close - The use of records to store data - The use of SQL to search for data - The use of arrays (or equivalent) when solving problems, including both one-dimensional and two-dimensional arrays - How to use sub programs (functions and procedures) to produce structured code - Random number generation

Half Term 3

2.1.2 Designing, creating and refining algorithms - Identify the inputs, processes, and outputs for a problem - Structure diagrams - Create, interpret, correct, complete, and refine algorithms using: Pseudocode, Flowcharts - Reference language/high-level programming language - Identify common errors - Trace tables
2.1.3 Searching and sorting algorithms - Standard searching algorithms: Binary search,

Linear search - Standard sorting algorithms: Bubble sort, Merge sort, Insertion sort **2.3.1 Defensive design** - Defensive design considerations: Anticipating misuse, Authentication, Input validation - Maintainability: Use of sub programs, Naming conventions, Indentation, Commenting
Half Term 4 **2.3.2 Testing** - The purpose of testing - Types of testing: Iterative, Final/terminal - Identify syntax and logic errors - Selecting and using suitable test data: Normal, Boundary, Invalid, Erroneous - Refining algorithms **Practical Programming Practice**
Half Term 5 **Practical Programming Practice** **2.5.1 Languages -** Characteristics and purpose of different levels of programming language: High-level languages, Low-level languages, The purpose of translators The characteristics of a compiler and an interpreter **2.5.2 The Integrated Development Environment (IDE)** - Common tools and facilities available in an Integrated Development Environment (IDE): Editors, Error diagnostics, Run-time environment, Translators
Half Term 6 Revision & Exam Technique in preparation for Mock Exam, Feedback from mock and consolidation, reflection of Unit 2 Intro and overview, and preparation for Unit 1