Definitions and Concepts for AQA Computer Science GCSE

Topic 2: Programming

Data Type: Defines the type/value of data (e.g., integer, string, boolean).

Integer: A data type representing whole numbers.

Real (Float): A data type representing numbers with fractional parts.

Boolean: A data type representing true or false values.

Character: A data type representing a single character.

String: A data type representing a sequence of characters.

Variable: A storage location that can hold different values during program execution.

Constant: A value that does not change during program execution.

Variable Declaration: A statement type used to name and optionally assign an initial value to a variable.

Constant Declaration: A statement type used to name and assign a fixed value to a constant.

Identifier Names: Names for variables, constants, and subroutines.

Assignment: A statement type used to give a value to a variable or constant.

Sequence: Executing instructions one after the other.

Selection: Using conditions to control the flow (e.g., IF statements).

Iteration: Repeating a set of instructions (e.g., WHILE, FOR loops).

Definite (Count Controlled) Iteration: Iteration where the number of repetitions is known in advance (e.g., FOR loop).

Indefinite (Condition Controlled) Iteration: Iteration where the number of repetitions is not known in advance, and execution continues until a condition is met (e.g., WHILE, REPEAT...UNTIL, DO...WHILE).

Nested Selection: A selection structure placed inside another selection structure.

Nested Iteration: An iteration structure placed inside another iteration structure.

Arithmetic Operator: Operators that perform calculations (e.g., +, -, *, /).

Relational Operator: Operators that compare values (e.g., ==, !=, >).

Boolean Operator: AND, OR, NOT used in logical conditions.

Subroutine: A named block of reusable code designed to perform a task.

Procedure: A subroutine that performs an action but doesn't return a value.

Function: A subroutine that returns a value.

Robustness: How well a program can handle errors or unexpected inputs.

Addition: An arithmetic operation.

Subtraction: An arithmetic operation.

Multiplication: An arithmetic operation.

Real Division: An arithmetic operation that produces a real number result.

Integer Division: An arithmetic operation that produces the integer quotient (whole part) of a division.

Remainder (Modular Arithmetic): The value left over when one integer is divided by another (e.g., 11 MOD 2 = 1).

Equal To (==): A relational operation.

Not Equal To (< > or !=): A relational operation.

Less Than (<): A relational operation.

Greater Than (>): A relational operation.

Less Than Or Equal To (<=): A relational operation.

Greater Than Or Equal To (>=): A relational operation.

NOT: A Boolean operation.

AND: A Boolean operation.

OR: A Boolean operation.

Data Structure: A concept referring to ways of organizing and storing data.

Arrays: A data structure that stores a collection of items, typically of the same data type, accessed using an index (one- and two-dimensional are required).

Records: A data structure used to group related data items of potentially different data types under a single name.

Length (String): An operation to determine the number of characters in a string.

Position (String): An operation to find the starting index of a substring within a string.

Substring: An operation to extract a sequence of characters (string) within a larger string.

Concatenation: An operation to join two or more strings together.

Random Number Generation: The ability to use functions to produce pseudo-random numbers within programs.

Parameters: Data passed to a subroutine.

Local Variables: Variables declared within a subroutine that only exist while the subroutine is executing and are only accessible within that subroutine.

Data Validation Routines: Simple routines that check the validity of data being entered by a user (e.g., minimum length, empty string check, range check).

Authentication Routines: Simple routines that use a username and password (plain text) to verify a user's identity.

Testing (Algorithms and Programs): The process of verifying that an algorithm or program works correctly and meets its requirements.

Test Data: Data designed and used to check if a program works as expected.

Normal (Typical) Test Data: Test data that falls within the expected or typical range of inputs.

Boundary (Extreme) Test Data: Test data that lies at the limits of valid input ranges, including values just inside and just outside the valid range.

Erroneous Test Data: Test data that is invalid or incorrect, used to check how the program handles unexpected inputs.

Syntax Error: An error in the grammar or rules of the programming language, preventing the code from being translated or run.

Logic Error: An error in the program's design or algorithm that causes it to produce incorrect or unexpected results, even if it runs without crashing.