Definitions and Concepts for OCR Computer Science A-level

Component 1.4: Data Types, Data Structures and Algorithms

1.4.1 Data Types

AND (∧): A logical operator which returns TRUE (or 1) if and only if all inputs are TRUE (or 1).

ASCII: A character set used to represent alphanumeric characters or symbols as a set of 8 bits.

Binary: A number system that only uses ones and zeros to represent numbers (a base 2 system).

Bitwise Manipulation: Operations performed on a set of bits.

Boolean: A data type that can only store one of two possible values(1 or 0, TRUE or FALSE etc.)

Character: A data type for storing a letter, number or special character.

Denary: A number system that only uses 10 characters (0 to 9) to represent numbers (a base 10 system).

Floating Point Arithmetic: Performing arithmetic operations on floating point numbers in binary.

Hexadecimal: A number system that only uses 16 characters (0 to 9 and A to F) to represent numbers (a base 16 system).

Integer: A data type for storing whole number values (positive or negative) with no decimal parts.

OR (∨): A logical operator which returns TRUE (or 1) if and only if any one of the inputs are TRUE (or 1).

Primitive Data Type: A basic built-in data type provided by a programming language.

Real/Floating Point: A data type for storing numbers with decimal or fractional parts.

Shifts: A bitwise manipulation where a set of bits are all moved by one place in a given direction, and the end bit may be taken as a carry or appended to the other end depending on the shift method.

String: A data type for storing a sequence of alphanumeric characters or symbols, typically within quotation marks.

Two's Complement: A method of storing negative numbers in binary. It involves flipping all the bits of the binary representation of the positive number and then adding 1.

UNICODE: A character set that is a superset of ASCII. It is used to represent alphanumeric characters or symbols as an integer code point which is equal to that character's ASCII code.

XOR: A logical operator which returns TRUE (or 1) if and only if exactly 1 of the inputs are TRUE (or 1).

1.4.2 Data Structures

Arrays: A data structure for storing a finite, ordered set of data of the same data type within a single identifier.

Binary Search Tree: A tree where each node cannot have more than 2 children. The right node and its descendents always have a greater value than the root node (first data item).

Breadth First Traversal: A method of traversing an entire graph by visiting all the neighbours of the first node before repeating the same with each neighbour in the order they were visited.

Depth First Traversal: A method of traversing an entire graph by travelling as far as possible along one route before backtracking and trying alternative unexplored routes.

Directed Graphs: A graph where the order of the vertices paired in an edge matter. The edges are one way.

Graphs: A data structure consisting of a set of vertices/nodes connected by a set of edges/arcs.

Hash Table: A data structure where a hashing algorithm calculates a value to determine where a data item is to be stored. The data item can then be directly accessed by recalculation, without any search.

Linked Lists: A data structure that stores an ordered sequence of data where each item is stored with a pointer to the next item. The items are not stored contiguously (in this same order) in memory.

Lists: A data structure that stores a sequence of data values, each with a unique index.

Queues: A first-in-first-out (FIFO) data structure. The first item added/pushed on to the queue is the first to be removed/popped off.

Records: A data structure that stores data in elements called fields, organised based on attributes.

Stacks: A last-in-first-out (LIFO) data structure. The last item added/pushed is the first to be removed/popped off.

Trees: A data structure that uses a set of linked nodes to form a hierarchical structure starting at a root node. Each node is a child/sub-node of a parent node.

Tuples: A data structure for storing an immutable (cannot be modified once created), ordered set of data, which can be of different data types, within a single identifier.

Undirected Graphs: A graph where the order of the vertices paired in an edge does not matter. The edges are bidirectional.

1.4.3 Boolean Algebra

Association Laws: $A \wedge (B \wedge C) = (A \wedge B) \wedge C$,

$$A \vee (B \vee C) = (A \vee B) \vee C$$

Boolean Expressions: A combination of boolean variables and logical operators which evaluates to either TRUE or FALSE depending on the input.

Boolean Logic: A type of algebra with logical operators where all values and expressions ultimately reduce to TRUE or FALSE.

Commutation Laws: $A \wedge B = B \wedge A$,

$$A \vee B = B \vee A$$

De Morgan's First Law: $\neg(A \vee B) = \neg A \wedge \neg B$

De Morgan's Second Law: $\neg(A \wedge B) = \neg A \vee \neg B$

Distribution Laws: $A \wedge (B \vee C) = (A \wedge B) \vee (A \wedge C)$,

$$(A \vee B) \wedge (C \vee D) = (A \wedge C) \vee (A \wedge D) \vee (B \wedge C) \vee (B \wedge D)$$

Double Negation Law: $A = \neg\neg A$

D-Type Flip Flops: A sequential logic circuit used to store a single bit. It has two stable states, which can be flipped between using an input signal.

Full Adders: A combination of two half adders that takes a carry bit and two other input bits and returns their sum and the new carry as two output bits.

Half Adders: A combinational arithmetic circuit that adds two numbers and produces a sum bit (S) and carry bit (C) as the output.

Karnaugh Maps: A method of simplifying boolean expressions by redrawing the truth table and applying a set of visual rules to obtain expressions with (or close to) the minimum logical operators, as a sum of products.

Logic Gate Diagrams: A graphical method of representing boolean expressions using the standard symbols for logic gates.