OCR A-Level Computer Science – Paper 2: Algorithms and Programming Lower 6th End of Year Exam (Component 02 H446)

1. Elements of Computational Thinking

1.1 Thinking Abstractly:

- The nature and need for abstraction.
- Differences between abstraction and reality.
- Creating abstract models for different scenarios.

1.2 Thinking Ahead:

- Identifying inputs and outputs.
- Determining preconditions for problem solutions.
- The need for reusable program components.
- Caching: nature, benefits, and drawbacks.

1.3 Thinking Procedurally:

- Identifying components of problems and solutions.
- Ordering solution steps and sub-procedures.

1.4 Thinking Logically:

- Decision points in solutions.
- Logical conditions affecting decisions and flow.

1.5 Thinking Concurrently:

- Identifying parts of a problem solvable concurrently.
- Benefits and trade-offs of concurrent processing.

---

2. Problem Solving and Programming

2.1 Programming Techniques:

- Constructs: sequence, iteration, branching.
- iteration.
- Local and global variables.
- Functions, procedures, parameter passing (by value/reference).

2.2 Computational Methods:

- Problem recognition and decomposition.
- Divide and conquer approach.
- Use of abstraction.

2.3 **Algorithms:**

- Analysis and design of algorithms.
- Comparing algorithm suitability based on data and task.
- Main algorithms for data structures:
    - Stacks (push, pop, peek), queues (enqueue, dequeue)
- Standard algorithms (including programming):
    - Sorting: bubble sort, insertion sort, merge sort
    - Searching: binary search, linear search

---

3. Required Programming Algorithms

- Sorting: Bubble sort, insertion sort, merge sort
- Searching: Binary search, linear search (implementation required).
- Data structure algorithms: Stacks (push, pop, peek), queues (enqueue, dequeue)

Additional Key Areas:

- **Analysis of Problems:** Identifying problems suitable for computational solutions.