

Paper 1: Computational Thinking and Programming Skills

1. **Representing algorithms:**
 - a. Pseudo-code and programming basics (3.1.1).
 - b. Assignment statements and data types (3.2.1, 3.2.2, 3.2.8).
 - c. String handling operations (3.2.8).
2. **Flow of control:**
 - a. Relational and logical operations (3.2.4, 3.2.5).
 - b. Iteration and selection (if/else structures) (3.2.2).
 - c. Modulus (remainder) operations (3.2.3).
3. **Programming concepts:**
 - a. Writing Python programs (e.g., user input and output) (3.2.7).
 - b. Writing test plans (3.2.11).
 - c. Error identification (syntax and logic errors) (3.2.11).
4. **Abstraction and decomposition:**
 - a. Definitions and examples of abstraction (3.1.1).
 - b. Breaking problems into sub-problems (decomposition) (3.1.1).
5. **Lists and arrays:**
 - a. Manipulating data structures (3.2.6).
 - b. Trace tables for algorithms (3.1.1).
6. **Records and data management:**
 - a. Record definitions (3.2.6).
 - b. Creating and comparing records (3.2.6).
7. **Subroutines and functions:**
 - a. Writing reusable Python code (3.2.10).
 - b. Nested functions and parameter passing (3.2.10).
8. **Validation and authentication:**
 - a. Writing Python programs for data validation (3.2.11).
 - b. Creating authentication routines (username/password) (3.2.11).
9. **Problem-solving with 2D arrays:**
 - a. Sliding puzzle logic (3.2.6).
 - b. Movement constraints and board representations (3.2.6).
10. **Games and simulation:**
 - a. Developing rule-based simulations (3.2.2, 3.2.6).
 - b. Error handling and iteration-based solutions (3.2.2).
11. **Search and sorting algorithms:**
 - a. Linear search mechanics (3.1.3).

12. Properties of local variables:

- a. Scope and differences compared to global variables (3.2.10).

Paper 2: Computing Concepts

1. Fundamentals of data representation:

- a. Binary, hexadecimal, and decimal conversions (3.3.2).
- b. Binary arithmetic and shifts (3.3.4).

2. Data representation for media:

- a. Bitmap image representation (3.3.6).
- b. File size calculations based on resolution and color depth (3.3.6).

3. Logic gates and Boolean expressions:

- a. Truth tables for AND, OR, NOT, XOR (3.4.2).
- b. Writing Boolean expressions from logic circuits (3.4.2).

4. Components of computer systems:

- a. CPU components: control unit, cache, clock (3.4.5).
- b. Performance improvement factors: cores, clock speed, cache size (3.4.5).

5. Data compression techniques:

- a. Huffman coding mechanics and efficiency (3.3.8).

6. Programming languages:

- a. High-level vs. low-level languages (3.4.4).
- b. Interpreters, compilers, and assemblers (3.4.4).

7. Computer networks:

- a. Network types: LAN, WAN, PAN (3.5).
- b. Protocols: TCP/IP model, HTTP, SMTP, IMAP (3.5).
- c. Advantages and disadvantages of wired vs. wireless networks (3.5).

8. Cybersecurity:

- a. Cybersecurity threats: malware, phishing, blagging (3.6.2).
- b. Methods to secure systems (e.g., firewalls, encryption) (3.6.3).

9. Relational databases and SQL:

- a. Table structure, primary/foreign keys (3.7.1).
- b. Writing SQL queries (3.7.2).

10. Ethical, legal, and environmental impacts:

- a. Data privacy and societal impacts of technology (3.8).