CS23101 COMPUTATIONAL THINKING

WEEK 5&6

Unit - II UNDERSTANDING DATA AND PATTERN RECOGNITION

Logical thinking – reasoning, Pattern recognition in data, data sequences, puzzles, nonograms. Data Encryption – ciphering sentences and Compression

Compiled by Dr.DEJEY

Logical Thinking and Reasoning

Logical thinking is the **ability to analyze information** systematically, **make inferences, and arrive at conclusions** based on reasoned arguments. It's a crucial skill in problem-solving, programming, mathematics, and everyday decision-making.

- It is the process of **drawing conclusions based on evidence.**
- It is a form of control thinking in which the **thought process is continuously towards the solutions** of a problem.

Logical Thinking and Reasoning

- Reasoning is the highest form of thinking **to find out causes and predict effects**.
- An individual tries to solve problem by incorporating two or more aspects of his **past experience**.

Types of Logical Thinking and Reasoning

- **Deductive Reasoning**: Starting with a general principle and deriving specific facts from it.
  - Example: All humans are mortal. Socrates is a human. Therefore, Socrates is mortal.

- **Inductive Reasoning**: Making broad generalizations from specific observations.
  - Example: Every swan we've seen is white, so all swans must be white.

Types of Logical Thinking and Reasoning

- **Abductive Reasoning**: Starting with an incomplete set of observations and proceeding to the likeliest possible explanation.
  - Example: The lawn is wet in the morning; it probably rained overnight.

Importance: Logical reasoning helps in making sound decisions, debugging code, analyzing data, and solving puzzles.

Pattern Recognition in Data

Pattern recognition involves identifying regularities or patterns in data. This skill is critical in fields like data science, AI, cryptography, and more.

- **Types of Patterns**:
  - **Repetitive Patterns**: Identifying repeated sequences.
  - **Trends**: Observing an upward or downward movement in data.
  - **Anomalies**: Spotting data points that deviate from the expected pattern.

Pattern Recognition in Data

Applications:

- **Data Analysis**: Identifying trends or outliers.
- **Machine Learning**: Recognizing patterns in training data to make predictions.
- **Cryptography**: Detecting patterns that could lead to breaking encryption.

Examples:

- **Visual Patterns**: Recognizing shapes or colors that repeat in images.
- **Mathematical Patterns**: Finding sequences like Fibonacci or arithmetic progressions.

Data Sequences and Puzzles

Data sequences involve an ordered collection of elements, often following a specific rule. Understanding these sequences is fundamental in mathematics, computer science, and logic-based puzzles.

- **Arithmetic Sequences**: Difference between consecutive terms is constant.
  - Example: 2, 4, 6, 8, ...
- **Geometric Sequences**: Each term is a fixed multiple of the previous term.
  - Example: 3, 9, 27, 81, ...

Data Sequences and Puzzles

Puzzles Involving Sequences:

- **Sudoku**: Fills a grid based on the logic of number sequences.
- **Magic Squares**: Arranging numbers in a square grid where the sums of each row, column, and diagonal are the same.

Problem-Solving Approach:

- **Identify the Rule**: Determine if the sequence follows a mathematical operation (addition, multiplication, etc.).
- **Predict the Next Element**: Use the identified rule to extend the sequence.

Nonograms

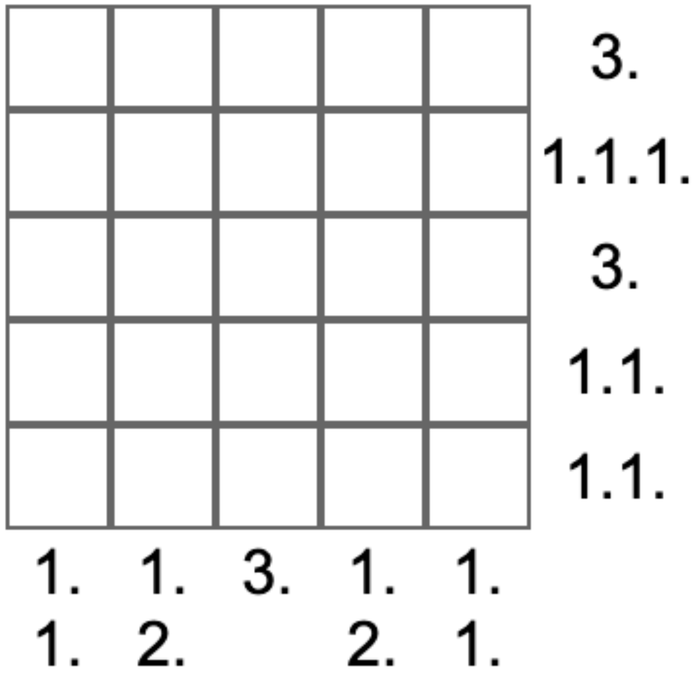
Nonograms, also known as "Picross" or "Griddlers," are logic puzzles where cells in a grid must be colored or left blank according to numbers at the side of the grid to reveal a hidden image.

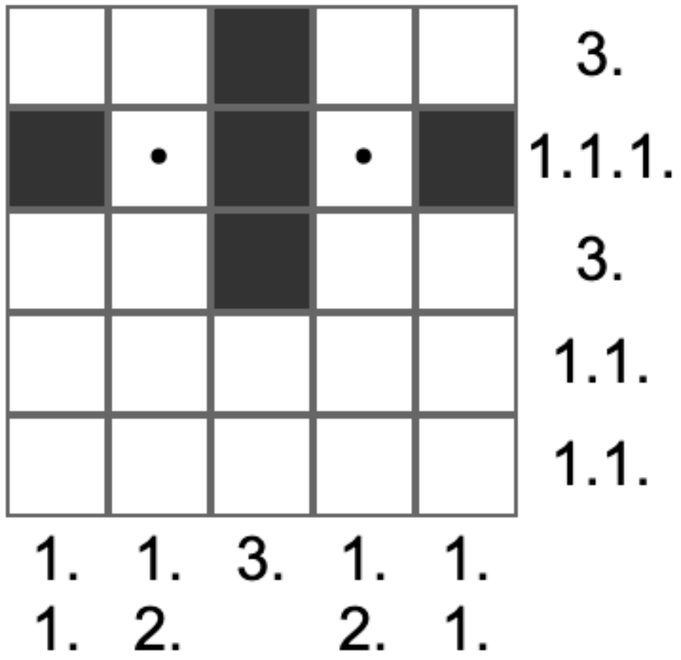
- **Rules**:
  - The numbers indicate the lengths of runs of consecutive filled cells.
  - There must be at least one empty cell between consecutive runs.

Nonograms

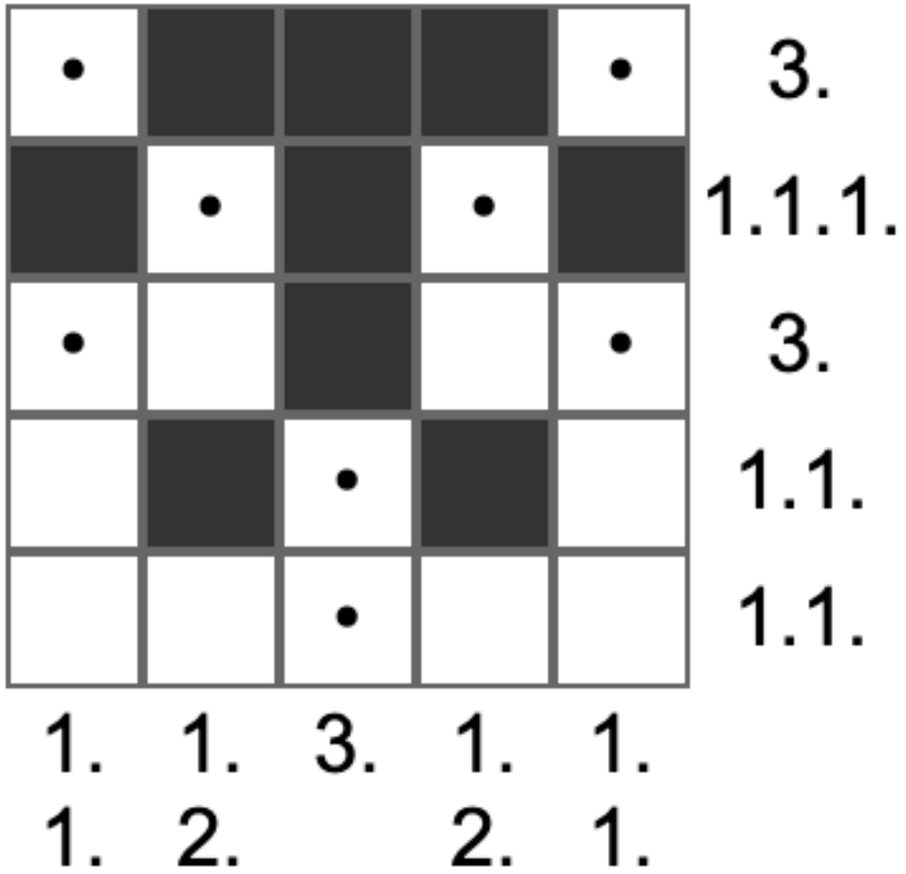
- **Solving Strategy**:
  - **Cross-referencing**: Compare rows and columns to reduce possible placements.
  - **Marking Certain Cells**: Fill in cells that are definitely part of a run.
  - **Use Logic**: Deduce which cells must be blank or filled based on the constraints.

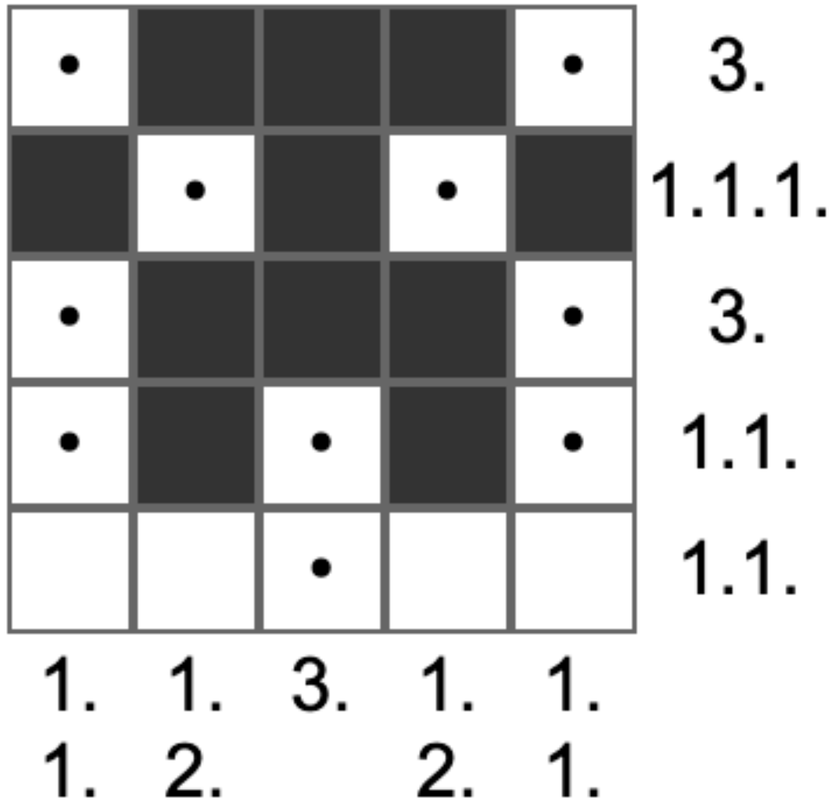
Nonograms



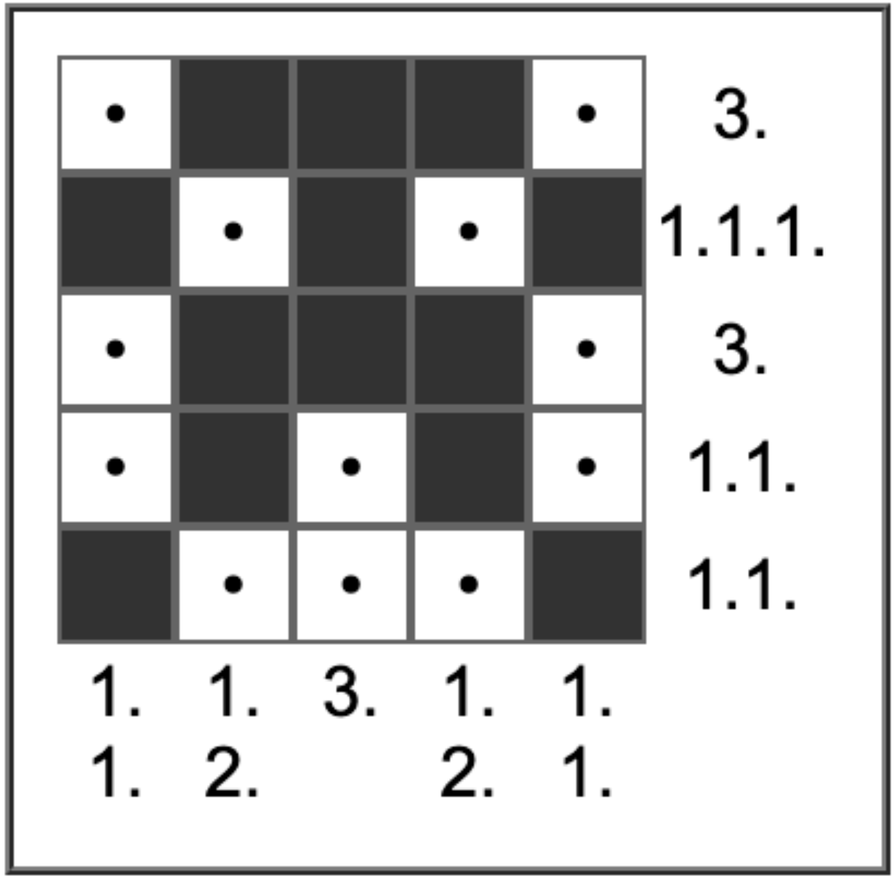


Nonograms





Nonograms



Data Encryption - Ciphering Sentences

Encryption is the process of converting data into a coded format to prevent unauthorized access.

- **Caesar Cipher**: A substitution cipher where each letter in the plaintext is **shifted a certain number of places down or up** the alphabet.
  - Example: With a shift of 3, A becomes D, B becomes E, etc.
- **Substitution Cipher**: Each letter in the plaintext is **replaced by a letter with some fixed relationship to it.**
  - **Vigenère Cipher**: Uses a keyword to shift letters based on the position of the keyword letters.

Data Encryption - Ciphering Sentences

Steps in Encryption:

- **Choose a Cipher**: Determine the encryption method (e.g., Caesar Cipher).
- **Apply the Cipher**: Encode the message using the chosen method.
- **Share the Key**: Provide the key for decryption (e.g., the shift value for Caesar Cipher).

Importance: Encryption ensures data privacy and security in communication.

Data Compression

Data compression reduces the size of data to save space or transmission time. It is a vital technique in computer science, particularly for storing and transmitting large files.

- **Lossless Compression**: Reduces data size without any loss of information.
  - Example: **Huffman Coding** - Assigns shorter codes to more frequent characters.
- **Lossy Compression**: Reduces data size by removing some information, often imperceptible to users.
  - Example: JPEG image compression reduces file size by discarding less important data.

Data Compression

Applications:

- **File Storage**: Compressing files to save disk space.
- **Data Transmission**: Reducing the bandwidth needed to transmit data over the internet.
- **Streaming**: Compressing audio and video to allow real-time playback.