Learning summary report – Complexity

Complexity Introduction

- Analyzing computational issues 'complexity and resource needs-particularly those related to time and space-is known as mathematical complexity.

Key concepts

- An algorithm is used to solve a computational problem that is defined by inputs and outputs.
- Abstract machines such as RAM machines and turning machines are examples of modules of computation.

Complexity Metrics

- Time complexity: Number of steps needed in relation to input size.
- Space complexity: Memory use based on input size

Asymptotic Notation

- Big O (O): Maximum bound, worst-case situation (e.g. O (n^2)).
- Omega (Ω): Lower limit, optimal situation (e.g. Ω (n)).
- Theta (Θ) represents the tight bound and growth rate (e.g.Θ (n log n))

Complexity Classes

- Class P (Polynomial Time)
- Class NP (Non – deterministic Polynomial Time)
- NP- Complete and NP-Hard

Reductions

- Proving using polynomial-time transformations that problem B. For instance, converting 3-SAT to TSP

Conclusion

- For both theoretical and practical applications, mathematical complexity offers a foundation for comprehending the bounds of computer problems and the effectiveness of algorithms.