

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <p><strong>Course Description:</strong> This course introduces basic elements
of the design and analysis of computer
    algorithms. Topics include asymptotic notations and analysis, divide and
conquer strategy, greedy methods,
    dynamic programming, basic graph algorithms, NP-completeness, and
approximation algorithms. For each topic,
    beside in-depth coverage, one or more representative problems and their
algorithms shall be discussed. </p>
  <ul>
    <strong>Course Objectives:<br></strong>
    <li>Analyze the asymptotic performance of algorithms.</li>
    <li>Demonstrate a familiarity with major algorithm design techniques
</li>
    <li>Apply important algorithmic design paradigms and methods of analysis.
</li>
    <li>Solve simple to moderately difficult algorithmic problems arising in
applications.</li>
    <li>Able to demonstrate the hardness of simple NP-complete
problems</li>
  </ul>

  <p>
    <strong>Foundation of Algorithm Analysis</strong><br>
1.1. Algorithm and its properties, RAM model, Time and Space Complexity, detailed
analysis
of algorithms (Like factorial algorithm), Concept of Aggregate Analysis <br>
1.2. Asymptotic Notations: Big-O, Big-Ω and Big-Θ Notations their Geometrical
Interpretation
and Examples. <br>
1.3. Recurrences: Recursive Algorithms and Recurrence Relations, Solving
Recurrences

```

(Recursion Tree Method, Substitution Method, Application of Masters Theorem)

Iterative Algorithms(4)

2.1. Basic Algorithms: Algorithm for GCD, Fibonacci Number and analysis of their time and

space complexity.

2.2. Searching Algorithms: Sequential Search and its analysis

2.3. Sorting Algorithms: Bubble, Selection, and Insertion Sort and their Analysis.

</p>

</html>

127.0.0.1:5501/index.html

Course Description: This course introduces basic elements of the design and analysis of computer algorithms. Topics include asymptotic notations and analysis, divide and conquer strategy, greedy methods, dynamic programming, basic graph algorithms, NP-completeness, and approximation algorithms. For each topic, beside in-depth coverage, one or more representative problems and their algorithms shall be discussed.

Course Objectives:

- Analyze the asymptotic performance of algorithms.
- Demonstrate a familiarity with major algorithm design techniques
- Apply important algorithmic design paradigms and methods of analysis.
- Solve simple to moderately difficult algorithmic problems arising in applications.
- Able to demonstrate the hardness of simple NP-complete problems

Foundation of Algorithm Analysis

- 1.1. Algorithm and its properties, RAM model, Time and Space Complexity, detailed analysis of algorithms (Like factorial algorithm), Concept of Aggregate Analysis
- 1.2. Asymptotic Notations: Big-O, Big-Ω and Big-Θ Notations their Geometrical Interpretation and Examples
- 1.3. Recurrences: Recursive Algorithms and Recurrence Relations, Solving Recurrences (Recursion Tree Method, Substitution Method, Application of Masters Theorem)

Iterative Algorithms(4)

- 2.1. Basic Algorithms: Algorithm for GCD, Fibonacci Number and analysis of their time and space complexity.
- 2.2. Searching Algorithms: Sequential Search and its analysis
- 2.3. Sorting Algorithms: Bubble, Selection, and Insertion Sort and their Analysis.