

<This Note Is An Example>

Computer Science - Introduction to Algorithms

Date: August 31, 2024

Instructor: Prof. Smith

Topic: Introduction to Algorithms

1. What is an Algorithm?

- An algorithm is a step-by-step procedure or formula for solving a problem.
- It is a sequence of instructions that are followed to achieve a specific goal or perform a specific task.
- Examples:
 - A recipe for baking a cake.
 - A method to solve a Rubik's cube.

2. Characteristics of a Good Algorithm

- **Correctness:** Should solve the problem correctly.
- **Efficiency:** Uses minimal resources (time, space).
- **Definiteness:** Each step is clear and unambiguous.
- **Finiteness:** Must terminate after a finite number of steps.
- **Input/Output:** Accepts input and produces output.

3. Why Study Algorithms?

- To improve problem-solving skills.
- To write efficient code that runs faster and uses fewer resources.
- Essential for technical interviews and competitive programming.

4. Basic Algorithm Design Techniques

- **Divide and Conquer:** Breaks a problem into smaller subproblems, solves them independently, and combines results. Example: Merge Sort.
- **Greedy Algorithms:** Make the best choice at each step. Example: Dijkstra's Algorithm.
- **Dynamic Programming:** Solves problems by breaking them down into overlapping subproblems. Example: Fibonacci Sequence using memoization.
- **Backtracking:** Tries different possibilities and backtracks upon failure. Example: Solving a maze.

5. Analyzing Algorithms

- **Time Complexity:** Measures how the running time of an algorithm increases with the input size.
 - Big O Notation: Represents the upper bound of an algorithm's running time (worst-case scenario).
 - Common complexities: $O(1)$, $O(\log n)$, $O(n)$, $O(n \log n)$, $O(n^2)$, etc.
- **Space Complexity:** Measures the amount of memory an algorithm uses relative to the input size.

6. Example: Algorithm to Find Maximum Number in a List

python

```
def find_max(arr):  
    max_value = arr[0]  
    for num in arr:  
        if num > max_value:  
            max_value = num  
    return max_value
```