Introduction to Algorithms - Lecture Notes

# Summary of Lecture

This lecture introduces the concept of algorithms, their historical context, and their modern applications.   
An algorithm is defined as a computational procedure or recipe that transforms inputs into outputs via a   
step-by-step process. The lecture emphasizes the historical roots of algorithms, notable contributors such as   
Al-Khwarizmi, Alan Turing, and Donald Knuth, and the pivotal role of algorithms in modern computing.  
It also outlines how algorithms underpin numerous real-world applications, from search engines to logistics.

# Key Concepts

1. \*\*Definition\*\*: An algorithm is a computational procedure that takes inputs, processes them, and produces outputs.  
2. \*\*Instance of a Problem\*\*: Plugging actual values into an algorithm's inputs creates a specific instance to solve.  
3. \*\*Historical Significance\*\*: Ancient algorithms include multiplication methods, Euclidean GCD, and the sieve of Eratosthenes.  
4. \*\*Key Figures\*\*:  
 - Al-Khwarizmi: Origin of the word "algorithm," contributions to arithmetic and algebra.  
 - Alan Turing: Defined computational procedures with the Turing machine.  
 - Donald Knuth: Authored \*The Art of Computer Programming\*.  
5. \*\*Modern Importance\*\*: Algorithms are critical for efficient computation, particularly after the invention of computers.  
6. \*\*Applications\*\*: Search engines (Google), GPS routing, air traffic control, logistics (FedEx), and matching systems (Stable Marriage Algorithm).  
7. \*\*Algorithm Properties\*\*: Step-by-step, unambiguous, finite, and efficient.

# Example Pseudocode: Multiplication Algorithm

MULTIPLY(m, n):  
 result ← 0  
 for i from 1 to n:  
 result ← result + m  
 return result

# Python Example: Multiplication Algorithm

def multiply(m, n):  
 result = 0  
 for \_ in range(n):  
 result += m  
 return result  
  
# Example usage  
print(multiply(121, 234)) # Should print the product of 121 and 234

# Real-World Applications of Algorithms

1. \*\*Search Engines\*\*: Use indexing, hashing, and ranking algorithms to quickly retrieve relevant results.  
2. \*\*GPS Navigation\*\*: Employs shortest path algorithms (e.g., Dijkstra's) to compute optimal routes.  
3. \*\*Air Traffic Management\*\*: Uses scheduling and assignment algorithms for gates, runways, and flight paths.  
4. \*\*Logistics\*\*: TSP (Travelling Salesperson Problem) for delivery routes to minimize cost and time.  
5. \*\*Resource Allocation\*\*: Stable Marriage Algorithm for matching residents to hospitals or organ donors to recipients.