**Real Life Examples of Algorithms**

## **Real-World Algorithms and Their Roots**

1. **Google PageRank Algorithm (Important) ()**(Graph Traversal + Priority Queue)
2. ***A Search Algorithm*\***  
   (Graph Traversal + Heaps)
3. **Dijkstra’s Algorithm**(Shortest Path + Priority Queue)
4. **Kruskal’s Algorithm**(Minimum Spanning Tree + Greedy)
5. **Prim’s Algorithm**(Minimum Spanning Tree + Priority Queue)
6. **Bellman-Ford Algorithm**(Shortest Path Algorithm)
7. **Floyd-Warshall Algorithm**(All Pairs Shortest Path + Dynamic Programming)
8. **Huffman Encoding Algorithm**(Greedy Algorithm)
9. **Knuth-Morris-Pratt (KMP) Algorithm**(String Matching)
10. **Rabin-Karp Algorithm**(String Matching + Hashing)
11. **Trie-based Autocomplete**(Tree Traversal + Graph Traversal)
12. **Travelling Salesman Problem (TSP) Approximation**(Branch and Bound + Greedy)
13. **Job Sequencing with Deadlines**(Greedy Algorithm)
14. **Topological Sorting in Build Systems (Gradle, Make)**(Topological Sort - DFS/Kahn’s)
15. **Dynamic Programming Sequence Alignment**(Fibonacci/LCS/Knapsack DP)
16. **Plagiarism Detection Systems**(KMP, Rabin-Karp, LSH)
17. **Sieve of Eratosthenes**(Number Theory Algorithm)
18. **RSA Key Generation (Prime Number Detection)**(Number Theory + Sieve)
19. **Monte Carlo Simulation**(Randomized Algorithm)
20. **Randomized Quick Sort**(Divide and Conquer + Randomized)
21. **Database Indexing (B-Trees, Binary Search)**(Tree Traversal + Binary Search)
22. **Median of a Stream (Two Heaps)**(Priority Queue + Heap)
23. **Spell Checker (Trie + DFS)**(Tree Traversal)
24. **Load Balancer Consistent Hashing**(Hashing Algorithm)
25. **Convex Hull Detection in GIS Systems**(Geometric Algorithm)
26. **Collision Detection in Gaming Engines**(Convex Hull + Geometric)
27. **Online Leaderboard (Binary Search + Heap)**(Sorting + Searching)
28. **Dependency Resolution (Package Managers)**(Topological Sort)
29. **DDoS Detection in Firewalls (Bloom Filter + Hashing)**(Hashing + Bit Manipulation)
30. **Facial Recognition Clustering (LSH + K-Means)**(Hashing + Greedy + Graph Traversal)