Outline

1. Some Hard Problems
2. P, NP and Co-NP
3. Polynomial Time Reductions and NP-Completeness
4. NP-Complete Problems
5. Dealing with NP-Hard Problems
6. Summary
7. Summary of Studies 2024 Fall

Important notations/algorithms

- Introduction:
    - Asymptotic analysis: O, Ω, Θ, compare the orders

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph
  - Two representations: adjacency matrix, linked lists

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph
  - Two representations: adjacency matrix, linked lists
  - Path, cycle, tree, directed acyclic graph, bipartite graph

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph
  - Two representations: adjacency matrix, linked lists
  - Path, cycle, tree, directed acyclic graph, bipartite graph
  - Connectivity problem: BFS and DFS algorithm

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph
  - Two representations: adjacency matrix, linked lists
  - Path, cycle, tree, directed acyclic graph, bipartite graph
  - Connectivity problem: BFS and DFS algorithm
  - Testing Bipartiteness problem: test-bipartiteness-BFS or test-bipartiteness-DFS algorithm

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph
  - Two representations: adjacency matrix, linked lists
  - Path, cycle, tree, directed acyclic graph, bipartite graph
  - Connectivity problem: BFS and DFS algorithm
  - Testing Bipartiteness problem: test-bipartiteness-BFS or test-bipartiteness-DFS algorithm
  - Topological Ordering problem: topological-sort algorithm

Important notations/algorithms

- Introduction:
  - Asymptotic analysis: O, Ω, Θ, compare the orders
  - Polynomial time (efficient algorithm), exponential time
- Graph Basics:
  - Undirected graph, directed graph
  - Two representations: adjacency matrix, linked lists
  - Path, cycle, tree, directed acyclic graph, bipartite graph
  - Connectivity problem: BFS and DFS algorithm
  - Testing Bipartiteness problem: test-bipartiteness-BFS or test-bipartiteness-DFS algorithm
  - Topological Ordering problem: topological-sort algorithm
- HW 1, Quiz 2-4, P1

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem
  - Interval Partitioning problem

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem
  - Interval Partitioning problem
  - Offline Caching problem

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem
  - Interval Partitioning problem
  - Offline Caching problem
  - Priority Queue: heap

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem
  - Interval Partitioning problem
  - Offline Caching problem
  - Priority Queue: heap
  - Huffman Code problem

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem
  - Interval Partitioning problem
  - Offline Caching problem
  - Priority Queue: heap
  - Huffman Code problem
- HW 2, Quiz 4-6, P1, Mid Exam I

Important notations/algorithms

- Greedy algorithms: safety strategy+self reduce
  - Box Packing problem
  - Interval Scheduling problem
  - Interval Partitioning problem
  - Offline Caching problem
  - Priority Queue: heap
  - Huffman Code problem
- HW 2, Quiz 4-6, P1, Mid Exam I
- Quiz 6: Job scheduling with deadline, clustering problem, Weighted scheduling problem

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
    - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
    - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
    - Counting inversions problem

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
  - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
  - Counting inversions problem
  - Selection problem

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
  - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
  - Counting inversions problem
  - Selection problem
  - Polynomial Multiplication problem

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
    - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
    - Counting inversions problem
    - Selection problem
    - Polynomial Multiplication problem
    - Recurrences: recursive-tree method and Master Theorem

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
  - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
  - Counting inversions problem
  - Selection problem
  - Polynomial Multiplication problem
  - Recurrences: recursive-tree method and Master Theorem
  - Fibonacci number problem

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
  - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
  - Counting inversions problem
  - Selection problem
  - Polynomial Multiplication problem
  - Recurrences: recursive-tree method and Master Theorem
  - Fibonacci number problem
- HW 3, Quiz 7-8 (in-class quiz), P1-2

Important notations/algorithms

- Divide-and-Conquer algorithms: Divide+Conquer+Combine
  - Sorting problem: merge-sort algorithm, quick-sort algorithm (and In-Place sorting algorithm)
  - Counting inversions problem
  - Selection problem
  - Polynomial Multiplication problem
  - Recurrences: recursive-tree method and Master Theorem
  - Fibonacci number problem
- HW 3, Quiz 7-8 (in-class quiz), P1-2
- Quiz 8: Modular Exponentiation Problem, Matrix Multiplication, Closest Pair, Convex Hull

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
    - Weighted interval scheduling problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
    - Weighted interval scheduling problem
    - Subset Sum problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
    - Weighted interval scheduling problem
    - Subset Sum problem
    - Knapsack problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)
  - Edit distance with insertions and deletions problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)
  - Edit distance with insertions and deletions problem
  - Edit distance with insertions, deletions and replacing problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
    - Weighted interval scheduling problem
    - Subset Sum problem
    - Knapsack problem
    - Longest common subsequence problem (LCS)
    - Edit distance with insertions and deletions problem
    - Edit distance with insertions, deletions and replacing problem
    - Shortest Path in Directed Acyclic Graph (DAG)

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)
  - Edit distance with insertions and deletions problem
  - Edit distance with insertions, deletions and replacing problem
  - Shortest Path in Directed Acyclic Graph (DAG)
  - Matrix Chain Multiplication problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)
  - Edit distance with insertions and deletions problem
  - Edit distance with insertions, deletions and replacing problem
  - Shortest Path in Directed Acyclic Graph (DAG)
  - Matrix Chain Multiplication problem
  - Optimum Binary Search Tree Problem

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)
  - Edit distance with insertions and deletions problem
  - Edit distance with insertions, deletions and replacing problem
  - Shortest Path in Directed Acyclic Graph (DAG)
  - Matrix Chain Multiplication problem
  - Optimum Binary Search Tree Problem
- HW 3-4, Quiz 9-10 (in-class quiz), P2, Mid Exam II

Important notations/algorithms

- Dynamic Programming algorithms: subproblem+recurrence relation+calculate from base case
  - Weighted interval scheduling problem
  - Subset Sum problem
  - Knapsack problem
  - Longest common subsequence problem (LCS)
  - Edit distance with insertions and deletions problem
  - Edit distance with insertions, deletions and replacing problem
  - Shortest Path in Directed Acyclic Graph (DAG)
  - Matrix Chain Multiplication problem
  - Optimum Binary Search Tree Problem
- HW 3-4, Quiz 9-10 (in-class quiz), P2, Mid Exam II
- Quiz 10: Longest Increasing Subsequence, Maximum Total Weight Independent Set, Shortest Path With Even Number of Vertices, Counting number of inverted 5-tuples

Important notations/algorithms

- Minimum spanning tree problem
    - Kruskal alg

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal
  - Prim's alg

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal
  - Prim's alg
- Minimum spanning forest problem

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal
  - Prim's alg
- Minimum spanning forest problem
- Shortest path problem

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal
  - Prim's alg
- Minimum spanning forest problem
- Shortest path problem
  - Dijkstra alg

Important notations/algorithms

- Minimum spanning tree problem
    - Kruskal alg
    - Reverse Kruskal
    - Prim's alg
- Minimum spanning forest problem
- Shortest path problem
    - Dijkstra alg
    - Bellman-Ford alg

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal
  - Prim's alg
- Minimum spanning forest problem
- Shortest path problem
  - Dijkstra alg
  - Bellman-Ford alg
  - Floyd-Warshall alg

Important notations/algorithms

- Minimum spanning tree problem
  - Kruskal alg
  - Reverse Kruskal
  - Prim's alg
- Minimum spanning forest problem
- Shortest path problem
  - Dijkstra alg
  - Bellman-Ford alg
  - Floyd-Warshall alg
- HW 4, Quiz 11, P2

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC
  - Circuit SAT $\leq_P$ 3-SAT

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC
  - Circuit SAT $\leq_P$ 3-SAT
  - 3-SAT $\leq_P$ IS

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC
  - Circuit SAT $\leq_P$ 3-SAT
  - 3-SAT $\leq_P$ IS
  - IS $=_P$ Clique

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC
  - Circuit SAT $\leq_P$ 3-SAT
  - 3-SAT $\leq_P$ IS
  - IS $=_P$ Clique
  - IS $=_P$ Vertex Cover

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC
  - Circuit SAT $\leq_P$ 3-SAT
  - 3-SAT $\leq_P$ IS
  - IS $=_P$ Clique
  - IS $=_P$ Vertex Cover
  - 3-SAT $\leq_P$ 3 Coloring

Important notations/algorithms

- Complexity Classes
  - P, NP, Co-NP
  - NP-complete, NP hard
- Polynomial Reduction
  - HC $\leq_P$ HP and HP $\leq_P$ HC
  - Circuit SAT $\leq_P$ 3-SAT
  - 3-SAT $\leq_P$ IS
  - IS $=_P$ Clique
  - IS $=_P$ Vertex Cover
  - 3-SAT $\leq_P$ 3 Coloring
- HW 4, Quiz 12-13