CMSC 451 Aug 31. 2026

Algorithm Design

Algorithm : a well-defined procedure that takes an input and produces an output.

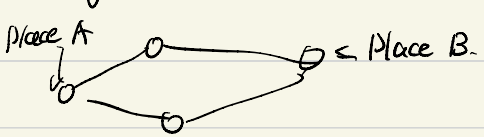
This course : maths / proofs, not codes

Algorithm Design : Given a problem (input & output)

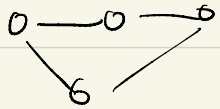
1. find an algorithm
2. prove that it is correct
3. analyze the cost of the algorithm.
   time complexity.

This class :

Graph Algorithm : Shortest Path.



Maps :



Social Networks -

Scheduling Problem :

task A : day 7 — 3 // 1 - 10

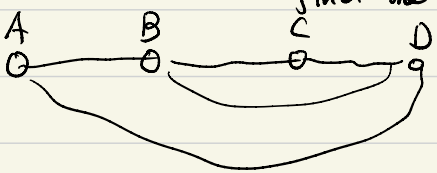
B : 2 — 5

C : 3 — 4

D : 2 — 8

problem : only one task each day

find the max # tasks you can do



2. Greedy Algorithms

Scheduling Algo : Always try the task that starts the earliest

not correct. → find a counter example

Always try the task that ends the earliest

correct. → prove that it is correct

3. Dynamic Programming

Value of a problem decomposed into value of smaller subproblems.

Maximum Interval Sum Problem

Input: $a_1, a_2, \cdots, a_n$.

Output: l, r, s.t. $a_l + \cdots + a_r$ is maximized.

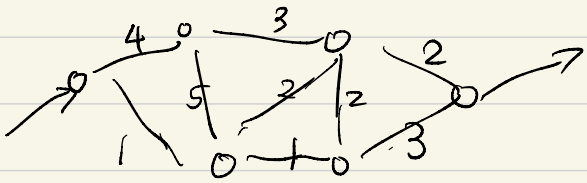
Algorithm: (DP) Define subproblem:

Problem #r { For any $r \in [1, \cdots, n]$
find l, s.t. $a_l + \cdots + a_r$ is maximized

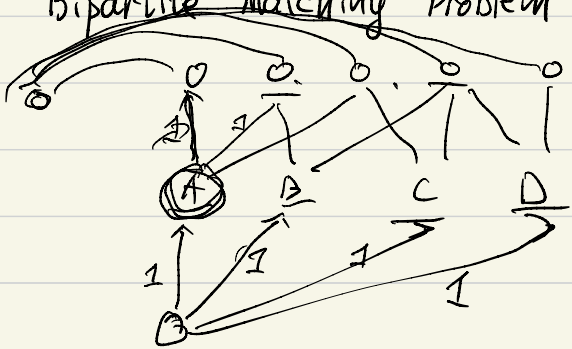
Problem #r $\Rightarrow$ Problem #r-1

1. drop all number $\# < r$
2. find the best l for Prob. #r-1 and add a_r to the solution

4. Network Flow.



Bipartite Matching Problem.



Items

Customers

5. Randomized Algorithms.

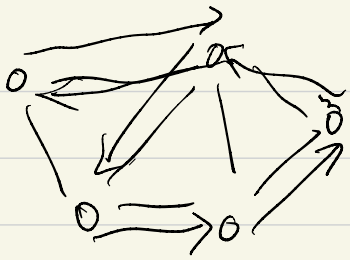
Quick Sort.

6. NP-Completeness.

Reduction: Algorithm for A → Algorithm for B

A is NPC : Algo for A → Algo for everything

7. Approximation Algorithms: Traveling Salesman Problem



IVPC.

Approx Algo : solution ≤ 2 · optimal solution