COMP-251A Review, Fall 2016

Lecture 1. (2 Sept 2016)

Course Outline.

Book Section 1.1 : Stable Matching

- Propose-and-reject algorithm. [Gale-Shapley 1962],
- Proof of correctness,
- Running time.

Lecture 2. (7 Sept 2016)

Book Sections 1.2 : Five Representative Problems

- Interval scheduling: n log n greedy algorithm.
- Weighted interval scheduling: n log n dynamic programming .
- Bipartite matching: n^k max-flow based algorithm.
- Independent set: **NP**-complete.
- Competitive facility location: **PSPACE**-complete.

Lecture 3. (12 Sept 2016)

Book Sections 2.1, 2.2, 2.4 : Analysis of Algorithms

- Polynomial time
- Worst case analysis
- Big-O notation and asymptotic growth.
- Simple analysis of loop based algorithms

Lecture 4. (14 Sept 2016)

Book Sections 5.1, 5.2 : Recurrence Relations

- Simple analysis of recursion based algorithms
- Mergesort
- Running time

Lecture 5-6. (19-21 Sept 2016)

Book Sections 5.4, 5.5: Divide-and-Conquer algorithms.

- Closest Pair of Points
- Matrix Multiplication

Book Section CLRS 4.3: Master Theorem.

- Master Method, Cases 1, 2, 3: comparing "$\log_b a$" to "f(n)".

Book Section BB 7.5: Median & Selection.

Lecture 7. (26 Sept 2016)

Book Sections 3.1, 3.2 : Introduction to graphs.

- Graph representation & Basic properties
- Connectivity — Breath-first search

Lecture 8. (28 Sept 2016)

Book Sections 3.4, 3.5, 3.6 : Introduction to graphs.

- Connected Components
- Testing Bipartiteness
- DAG and Topological ordering.

Lectures 9-10. (3-5 Oct 2016)

CLRS Book Chapters 12-13 : Red-Black Trees

- BST
- SEARCH, MINIMUM, MAXIMUM, PREDECESSOR, SUCCESSOR, INSERT, and DELETE
- Red-black property
- Operations on RB-T
  - RB-Tree-insert and RB-insert-fixup
  - RB-Tree-Delete and RB-delete-fixup
- Running times
- First repetition from RB Trees.

Lecture 11. (12 Oct 2016)

Book Sections 4.1, 4.2 : Greedy Algorithms.

- Basics of greedy algorithms
- Interval Scheduling
- Interval Partitioning
- Minimizing Lateness
- Running times
- Greedy Strategies

Lecture 12. (October 17th, 2016) MID-TERM

Lecture 13. (October 19th, 2016)

Book Chapter 4.4 : Greedy - Dijkstra.

- Weight of edges and paths
- Dijkstra's algorithm
- Running time analysis

Lecture 14. (October 24th, 2016)

Book Chapter 4.5, 4.7 : Greedy - Minimum spanning tree.

- Definition
- Generic MST algorithm
- Finding *safe* edges
- Cycles, Cuts, edge crossing, light edges,…
- Prim's algorithm and priority queues
- Kruskal's algorithm and Data structure for disjoint sets
- Running times
- Clustering

Lecture 15. (October 26th, 2016)

Book Chapter 4.6 : Data Structure for Disjoint Sets.

- Finding connected components in a graph
- Disjoint sets operations
  - ✶Make-set
  - ✶Union
  - ✶Find-set
- Link list representation
- Running times
- Forest representation
- Union by rank and path compression
- Running times

Lectures 16-17. (October 31st-November 2nd, 2016)

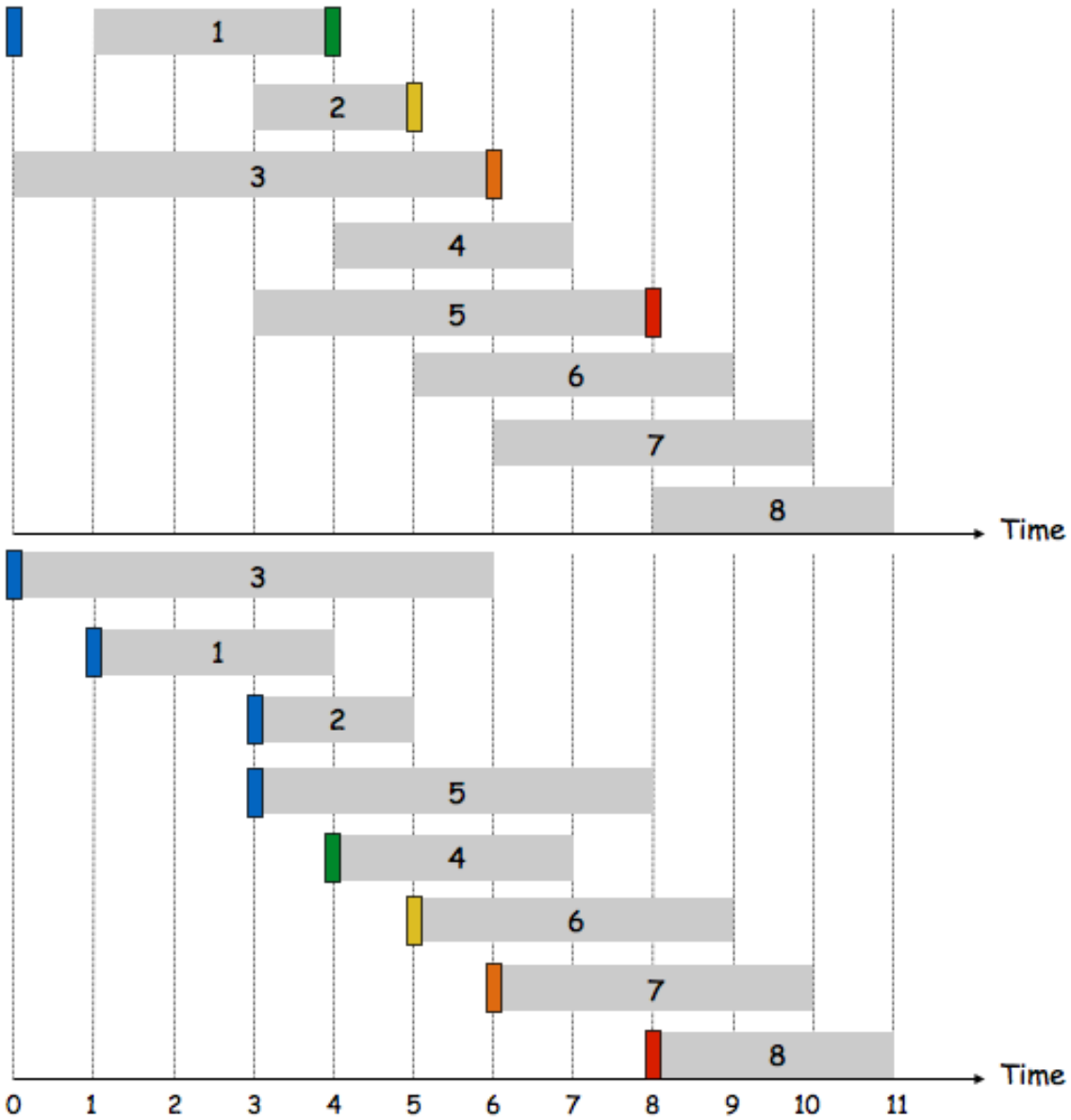
Book Sections 6.0, 6.1, 6.8, 6.10, 6.3, 6.5, 6.6, 6.7 : Dynamic Programming.

- Avoiding exponential time recursions
- Optimal sub-structure principle
- Weighted Interval Scheduling – Memoization
- Shortest paths again – Bellman-Ford algorithm
- Negative-weight edges/cycles
- Segmented Least Squares
- RNA Secondary Structure

Lecture 18. (November 7th, 2016)

Book Sections 6.6, 6.7 : Dynamic Programming Applications.

- Sequence alignment
- Running time (and space)
- Sequence alignment in linear space
- Running times (and space)


1
2
3
4
5
6
7
8
Time
3
1
2
5
4
6
7
8
Time
0
1
2
3
4
5
6
7
8
9
10
11

Lectures 19-20. (November 9th-14th, 2016)

Book Sections 7.1, 7.2, 7.3 : Max Flow and Min Cut.

- Minimum Cut Problem, Cut Capacity
- Maximum Flow Problem, Flow Value
- Flow Value ≤ Cut capacity
- Certificate of Optimality

Lectures 21-22. (November 16th-21st, 2016)

Book Sections 7.5, 7.6, 7.12, 7.10, 7.7, 7.8 : Max Flow and Min Cut + Applications.

- Residual Graph & Ford-Fulkerson Algorithm
- Max-Flow Min-Cut Theorem
- Choosing Good Augmenting Paths – capacity scaling
- Running times
- Applications
    1. Bipartite Matching
    2. Edge Disjoint Paths – Network Connectivity
    3. Baseball Elimination
    4. Image Segmentation
    5. Project Selection
    6. Extensions to Max Flow – Circulations with Demands (and Lower Bounds)
    7. Survey Design

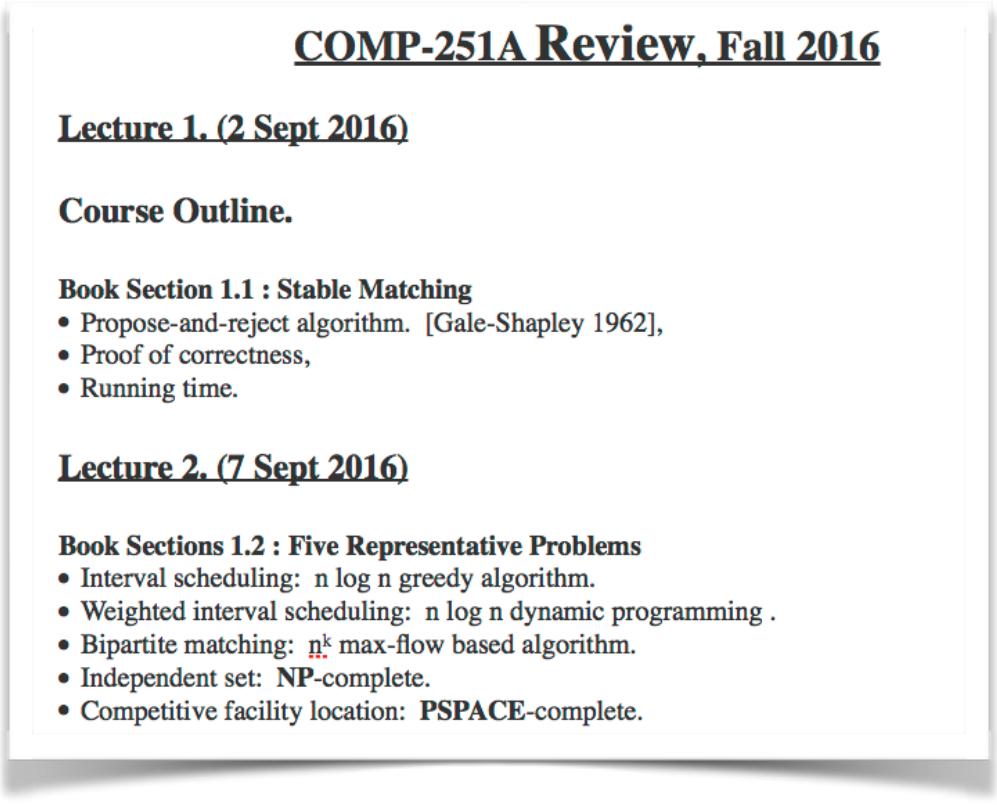
Lectures 23-24-25.(November 23rd-28th-30th, 2016)

Book Sections 13.1, 13.2, 13.3, 13.5, 13.6, 13.99 13 : Randomized Algorithms + Hashing.

- Randomized algorithms,
- Contention Resolution
- Global Minimum Cut
- Randomized Quick Sort
- Hashing and Universal Hashing.
- Introduction to Cryptography + Primality Testing

Lecture 26. (Dec. 5, 2016)

Review of course material in preparation to FINAL EXAM !



COMP-251A Review, Fall 2016

Lecture 1. (2 Sept 2016)

Course Outline.

Book Section 1.1 : Stable Matching

- Propose-and-reject algorithm. [Gale-Shapley 1962],
- Proof of correctness,
- Running time.

Lecture 2. (7 Sept 2016)

Book Sections 1.2 : Five Representative Problems

- Interval scheduling: n log n greedy algorithm.
- Weighted interval scheduling: n log n dynamic programming .
- Bipartite matching: n^k max-flow based algorithm.
- Independent set: **NP**-complete.
- Competitive facility location: **PSPACE**-complete.

COMP	Sec	Title	Date	Time	Prof.	FROM	TO	Bldg	Room	Row
COMP 251	001	Algorithms and Data Structures	Dec 13	2 pm	Crepeau	AAA	- ZZZ	GYM	Fieldhouse	26-31

Faculty of Science
Final Examination

Computer Science COMP-251A
Data Structures and Algorithms

Examiner: Prof. Claude Crépeau **Date:** Dec. 13, 2016

Associate Examiner: Prof. Yang Cai **Time:** 14:00 – 17:00

INSTRUCTIONS:

- This examination is worth 50% of your final grade.
- The total of all questions is 100 points.
- Each question is assigned a value found in brackets next to it.

- O P E N • B O O K S •/• O P E N • N O T E S

- Faculty standard calculator permitted only.
- This examination consists of 4 pages including title page.
- This examination consists of 4 questions.

SUGGESTION : read all the questions and their values *before* you start.

1)

[10%] **A)** Argue…

[5%] **B)** How many…

[10%] **C)** Given…

[10%] **D)** Analyze the running-time …

Use without proof:

2)

You are given…

A) We learned…

[10%]

B) Consider…

[5%]

Show…

3)

Consider

[10%] Explain...

4)

Consider…

[10%] (a) Draw…

[10%] (b) What…

[10%] (c) Draw…

[10%] (d) What...