Syllabus

<table>
<tr><th>Department</th><td>Math & CS</td><th>Credits</th><td>3</td><th>Instructor</th><td>Han-Saem Yun</td><th>Class Room</th><td>5708</td></tr>
<tr><th rowspan="2">Subject</th><td rowspan="2">MC422:
Algorithms</td><th rowspan="2">Class
hrs/wk</th><td rowspan="2">3</td><th colspan="2">Lab (e-mail)</th><th colspan="2">Attendee</th></tr>
<tr><td colspan="2">5707</td><td colspan="2"></td></tr>
</table>

1 Course Description

Course Objectives

On completion of this course, you'll hopefully be able to:

- Model/analyze real-world problems and design algorithmic solutions to the problems along with mathematical proofs;
  - using discrete structures and combinatorial methods
- Think computationally/algorithmically;
  - appreciate the importance of algorithms in CS and beyond;
  - understand/appreciate limits of computation.
  - learn/remember some basic tricks, algorithms, problems.

Contents

- Algorithm design patterns for tractable problems
  - Greedy strategy
  - Divide-and-conquer
  - Dynamic programming
  - Network flow (& duality/reduction)
  - Randomization
- Intractability (algorithm design anti-patterns)
  - NP-completeness (& reduction)
  - PSPACE-completeness
  - Undecidability
- Algorithm design patterns for intractable problems
  - Subproblem analysis (e.g. fixed-parameter tractability)
  - Approximation (e.g. pricing, LP relaxation, rounding/trimming for FPTAS)
  - Randomization
- Algorithms for online problems

2 Text & References

Text: "*Algorithm Design*", J. Kleinberg and E. Tardos

Reference: "*Introduction to Algorithms*", T. Cormen, C. Leiserson, and R. Rivest

Course online:

- http://ebook.ksa.hs.kr ⇒ New VOD ⇒ Log in
  ⇒ 정보과학 ⇒ 윤한샘 ⇒ MC422: Algorithms

3 Grading

- Grading table

Activities	Percentages
Problem Sets	25%
Midterm	15%
Final exam	50%
Attendance	10%

- Absolute evaluation
- Problem set for each week
- Miderm/Final exam: open-book/note, 5 hrs
- Late-work policy: −30%/day

4 Lecture Schedule

- 1: Stable Matching (1)
- 2-7: Intractability: NP (6)
- 8-12: Greedy algorithms (5)
- 13-15: Divide-and-conquer (3)
- 16-20: Dynamic programming (5)
- 22-27: Network Flow (6)
- 28: Intractability: PSPACE (1)
- 29-31: Subproblem analysis (3)
- 32-35: Approximation algorithms (4)
- 36-39: Randomized algorithms (4)
- 40: Online algorithms (1)

Introduction (1)

1. Stable matching
    - Gale-Shapley algorithm
    - Men-optimality, women-pessimality
    - Weakly pareto optimality
    - Strategic issues

Intractability: NP (6)

2. Polynomial-time reductions
    - Formal definition of algorithmic problems
    - Reduction by simple equivalence
    - Reduction from special case to general case (restriction)
    - Reduction by component design
3. NP-completeness
    - Definition of NP
    - Cook-Levin's theorem
    - NP-hardness & NP-completeness
4. NP-complete sequencing problems
    - 3-SAT $\leq_p$ directed Hamiltonian cycle $\leq_p$ Hamiltonian cycle $\leq_p$ TSP
    - 3-SAT $\leq_p$ longest simple path
5. NP-complete partitioning problems
    - 3-SAT $\leq_p$ 3-dimensional matching
    - 3-SAT $\leq_p$ 3-colorability $\leq_p$ 3-colorability on planar graphs
    - Planar k-colorability
6. NP-complete numerical problems
    - 3-SAT $\leq_p$ subset sum
    - 3-DM $\leq_p$ subset sum
    - Subset sum $\leq_p$ partition
    - Subset sum $\leq_p$ scheduling with release time
7. co-NP
    - Asymmetry of NP
    - Primality testing & factoring

Greedy algorithms (5)

8. Greedy algorithms for simple problems (exchange argument)
    - Selecting items to maximize total weights
    - Cashier's algorithm for coin changing
    - Truck driver's algorithm to select breakpoints

9. Greedy scheduling algorithms
   - Interval scheduling (greedy algorithm stays ahead)
   - Interval partitioning (structural property of solution)
   - Scheduling to Minimize lateness (exchange argument)
   - Farthest-in-future (FF) policy for offline caching
10. Shortest path
    - Dijkstra's algorithm for single-source shortest-path problem
11. Minimum spanning tree (MST)
    - Prim/Kruskal's algorithm for the MST problem
    - k-clustering
12. Huffman's greedy encoding for optimal prefix code

Divide-and-conquer (3)

13. Divide-and-conquer algorithms in discrete geometry, etc
    - Closest pair of points
    - Counting inversions using merge sort
14. Divide-and-conquer algorithms for multiplications
    - Karatsuba's algorithm for integer multiplication
    - Strassen's algorithm for matrix multiplication
15. Fast Fourier Transformation (FFT)
    - FFT for convolutions & polynomial multiplications

Dynamic programming (5)

16. Dynamic programming strategies: binary choice & multi-way choice
    - Binary choice: weighted interval scheduling
    - Multi-way choice: segmented least squares
17. Dynamic programming pattern: adding a new variable
    - Pseudo-polynomial time algorithm for Knapsack problem
18. Dynamic programming over intervals
    - CYK parsing
    - RNA secondary structure
19. Edit distance
    - Sequence alignment
    - Sequence alignment in linear space
20. Dynamic programming algorithms for the shortest-path problem
    - Bellman-Ford algorithm
    - Negative cycles
    - Distance vector protocol

21. *Wrap-up for the midterm*

Network Flow (6)

22. The maximum-flow problems
    - Weak duality of flows and cuts
    - Ford-Fulkerson method
    - Max-flow min-cut theorem
    - Flow Integrality
23. Algorithms for the maximum flow problem
    - Capacity scaling method
    - Preflow-push algorithm
24. Bipartite matching
    - Reduction from bipartite matching to max flow
    - Hall's theorem for perfect matching (c.f. Dilworth's theorem)
25. Reductions to max flow
    - Circulation with demands: reduction to max flow
    - Survey design: reduction to circulation with demands
    - Edge-disjoint paths: reduction to max flow
    - Network connectivity: duality with edge-disjoint paths
26. Reductions to min cut
    - Image segmentation: reduction to min cut
    - Project selection: reduction to min cut
    - Open pit mining: reduction to project selection
27. Baseball elimination

Intractability: PSPACE (1)

28. PSPACE
    - Quantified satisfiability
    - Planning problems
    - PSPACE-completeness

Subproblem analysis (3)

29. Subproblem analysis: vertex cover
    - Fixed-parameter tractability
    - Vertex cover in bipartite graph (König-Egerváry theorem from max-flow min-cut)
30. Subproblem analysis: independent set
    - independent set on trees (greedy algorithm)
    - weighted independent set on trees (dynamic programming)

31. Subproblem analysis: circular arc coloring
    - Fixed-parameter tractability (based on interval partitioning)

Approximation algorithms (4)

32. Greedy approximation algorithm
    - List scheduling for load balancing
    - Center selection
33. The pricing method
    - Vertex cover
34. LP relaxation
    - LP (linear programming) and ILP (integer linear programming)
    - Vertex cover
    - Generalized load balancing
35. Rounding/trimming scheme)
    - FPTAS (fully polynomial time approximation)
    - Knapsack
    - Subset sum

Randomized algorithms (4)

36. Analysis of randomized algorithms
    - Contention resolution
    - Karger's contraction algorithm for global min cut
37. Linearity of expectation
    - MAX 3-SAT
    - Monte Carlo (RP) vs. Las Vegas algorithms (ZPP)
38. Universal hasing
39. Chernoff bounds
    - Analysis of load balancing

Online algorithms (1)

40. Online algorithms
    - Competitiveness, strong competitiveness
    - LRU for fixed-space online caching
    - VMIN for variable-space online caching
    - Randomized online caching algorithm

5 Lecture Schedule (Tabular)

Lec #	Topics	Assignments	Categories
1	Stable Matching		Ch.1
2*	Poly-Time Reductions		Ch.8 Intractability
3*	NP-Completeness		
4	Hamiltonian Cycle, TSP	PS #1 due	
5	3D Matching, Graph Coloring		
6	Subset Sum, Knapsack	PS #2 due	
7	co-NP		
8	Exchange Argument		Ch.4 Greedy Algo.
9	Scheduling Problems		
10*	Shortest Path, Clustering	PS #3 due	
11*	Minimum Spanning Tree (MST)		
12	Optimal Prefix Code		
13	Closest Pair of Points	PS #4 due	Ch.5 Divide-Conquer
14	Integer/Matrix Multiplication		
15	Fast Fourier Transform (FFT)		
16	Binary/Multiway Choice	PS #5 due	Ch.6 Dynamic Prog.
17	Adding New Variable (Knapsack)		
18	DP over Intervals (CYK, RNA)		
19	Edit Distance	PS #6 due	
20	Shortest Path		
21	*Wrap-up for the midterm*	PS #7 due	
22	Max-Flow/Min-Cut Duality		Ch.7 Network Flow
23	Ford-Fulkerson, Capacity Scaling		
24	Bipartite Matching	PS #8 due	
25	Reductions to Max Flow		
26	Reductions to Min Cut		
27	Baseball Elimination	PS #9 due	
28	PSPACE-Completeness		Ch.9 PSPACE
29	Vertex Cover		Ch.10 Subproblems
30	Independent Set	PS #10 due	
31	Circular Arc Coloring		
32	Greedy (Load Balancing, Center)		Ch.11 Approximation
33	Pricing Method (Vertex Cover)	PS #11 due	
34	LP Relaxation (VC, Generalized LB)		
35	Rounding/Trimming for FPTAS		
36	Global Min Cut (Contraction Algo)	PS #12 due	Ch.13 Randomization
37	Linearity of Expectation (MAX 3-SAT)		
38	Universal Hashing		
39	Chernoff Bounds (Load Balancing)	PS #13 due	
40	Online Algorithms (Caching)		
41	*Wrap-up for the final exam*	PS #14 due	

*: overlap with MC322