CS091M4041H: Algorithm Design and Analysis -- Fall 2017

Course Information

- *Staff*
  Instructor:
  Dongbo Bu
  Email: dbu AT ict.ac.cn
  Location: Room 844, ICT Building,
  Tel: 62600844

  TAs:
  Jianwei Zhu, Lupeng Kong, Xiaoran Cao, Jingwei Zhang Fusong Ju, Guozheng Wei, Qi Zhang
  Email: TA of alGorithm Courses
  Location: 0817, ICT Building,
  Tel: 62600887
  Office Hours: 3:00pm-6:00pm, Wednesday

- *Textbooks (recommended, not required):*
  * T.H. Cormen, C.E. Leiserson, R. Rivest, and C. Stein, Introduction to algorithms (2nd ed.), MIT Press, 2001. Widely available.
  * J. Kleinberg and E. Tardos. Algorithm Design, Addison-Wesley, 2005.
  * R. Motwani and P. Raghavan, Randomized Algorithms. Cambridge U. Press, 1995
  * Christos H. Papadimitriou, Kenneth Steiglitz, Kenneth Steiglitz. Combinatorial Optimization: Algorithms And Complexity. Courier Dover Publications, 1998
  * Ding-Zhu Du, Ker-I Ko, Xiaodong Hu. Design and analysis of approximation algorithms. Springer, 2012
  * Daming Zhu, Shaohan Ma. Algorithm design and analysis. High Education Press, 2009.

  Other reading material:
  * Udi Manber, Introduction to Algorithms: A Creative Approach.
  * M. Mitzenmacher and E. Upfal, Probability and Computer. Cambridge U. Press, 2005.
  * M. R. Garey and D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. W.H. Freeman, New York, 1979.

- *Goals*:
  * to master the ability to extract mathematically clean core of a problem,
  * then identify an appropriate algorithm design technique based on the problem structure observations,
  * and analyze algorithm performance.

- *Prerequisites*:

We will assume knowledge of:
* Basic data structures such as lists, trees, heaps, sorting and searching
* Basic discrete mathematics such as proofs by mathematical induction;
* Computability and programming experience;

- *Topics*:
  We will cover the following topics if time permits.
  * Problem hardness, NP-completeness;
  * Algorithm analysis techniques, including worst-case and average-case, amortized, randomization, and competitive;
  * Basic algorithm techniques, including greedy, iteration, divide-and-conquer, dynamic programming, network flow, linear programming;
  * Algorithm techniques for hard problems, including approximation algorithms, local search, primal-dual algorithms, linear programming;
  * Randomized algorithms: basic techniques from discrete probability, and applications to optimization.
  * Specific problems from computational biology and Bioinformatics (if time permits).

Grading policies

Each student is expected to do 8 assignments and attend the final examination.

Weekly Schedule

The week number is an active link -- each week has its own page that includes required reading, recommended reading, assignment (if any), teaching assistants, etc. (Topics for weeks beyond the current and next are always tentative.)

- Week 1,2,3: Introduction to algorithm and basic design techniques
  - Lecture 1: Introduction to algorithm: some representative problems;
    Lec1.pdf ;
  - Lecture 3: Divide-and-conquer technique, and the combination with randomization;
    Lec5.pdf ; Lec5-FFT.pdf ; demo merge (by K. Wayne) ; demo merge inversion (by K. Wayne)
  - Reading material:
    - Chapter 1 of Algorithm design,
    - Chapter 17 of Introduction to Algorithms,
    - Lecture 8, 9 of The Design and Analysis of Algorithms,
    - On the solution of linear recurrence equations (by Mohamad Akra and Louay Bazzi, 1998) ,
    - College Admissions and the Stability of Marriage (by Gale and Shapley, 1962) ,
    - STABLE ALLOCATIONS AND THE PRACTICE OF MARKET DESIGN (compiled by the Economic Sciences Prize Committee of the Royal Swedish Academy of Sciences, 2012) ,
    - Stable matching: Theory, evidence, and practical design (INFORMATION FOR THE PUBLIC, The Nobel prize in economic sciences, 2012) ,

        - Who is Interested in Algorithms and Why? Lessons from the Stony Brook Algorithms Repository (by Steven S. Skiena, 1999) ,
        - Gene coexpression measures in large heterogeneous samples using count statistics (by Y. Wang, M. S. Waterman, and H. Huang, 2014) ,
        - Chapter 2,15,16,7,33.4 of Introduction to Algorithms,
        - Chapter 5,4,6 of Algorithm design,
        - Duality applied to the complexity of matrix multiplications and other bilinear forms (by J. Hopcroft, and J. Musinski, 1973) ,
        - The Coppersmith-Winograd matrix multiplication algorithm (by M. Anderson and S. Barman, 2009) ,
        - Some techniques for solving recurrences (by George S. Lueker, 1980)
        - Karatsuba algorithm vs. grade-school method: experimental results (by Carl Burch)
        - Fast Division of Large Integers --- A comparison of Algorithms (by Karl Hasselstrom, 2003)
        - Quicksort (by C. A. R. Hoare, 1962)
        - Ph. D. thesis of Michael Ian Shamos (Section 6.2)
        - Finding and counting given length cycles (by Noga Alon, Raphael Yuster, and Uri Zwick, 1994)
        - Closet Pair Data Structure: Applications (by David Eppstein)
        - Dynamic Euclidean Minimum Spanning Trees and Extrema of Binary Functions (by David Eppstein, 1995)
        - Fast Hierarchical Clustering and Other Applications of Dynamic Closest Pairs (by David Eppstein, 1998)
        - Fourier analysis (by Cleve Moler)
        - The complexity of computations (by A. A. Karatsuba, 1995) ,
        - Sorting and selection on dymamic data (by Aris Anagnostopoulos, et al, 2011) ,

- Week 3, 4, 5: More on basic algorithm design techniques;
    - Lecture 4: Dynamic programming technique;
      Lec6.pdf ; RNA secondary structure prediction (by Sarah Aerni) ; Edit distance (by Andrew McCallum); Publick-key cryptosystem (by Charles Clancy)
    - Reading material:
        - Chapter 2,15,16,7,33.4 of Introduction to Algorithms,
        - Chapter 5,4,6 of Algorithm design,
        - Gaussian eliminination is not optimal (by V. Strassen, 1968)
        - On Efficient Computation of Matrix Chain Products (by S. S. Godbole)
        - An O(n log n) algorithm for computation of matrix chain products (by T. C. Hu and M. T. Shing, 1981)
        - A general method applicable to the search for similarities in the amino acid sequence of two proteins (by Saul B. Needleman, Christian D. Wunsch, 1970)
        - Identification of Common Molecular Subsequences (by T.F.SMITH and M. S. WATERMAN, 1981)
        - The statistical distribution of nucleic acid similarities (by T.F.SMITH, M. S. WATERMAN, and C. Burks, 1985)
        - Esitmating statistical significance of sequence similarities (by M. S. WATERMAN, 1994)
        - Implementation of the Smith-Waterman Algorithm on a Reconfigurable Supercomputing Platform

        - A linear space algorithm for computing maximal common subsequences (by D. S. Hirschberg, 1975)
        - Basic Local Alignment Search Tool (by S. Altschul, et al. 1990)
        - P-value calculation (by J. Zhang)
        - PAM matrix for BLAST algorithm (by C. Alexander, 2002)
        - On a routing problem (by Bellman Ford, 1958)
        - Richard Bellman on the birth of dynamic programming (by Stuart Dreyfus, 2002)
        - Computing Partitions with Applications to the Knapsack problem (by E. Horowitz and S. Sahni, 1974)
        - The rise and fall of Knapsack cryptosystems (by A. M. Dolyzko)
        - A dynamic programming approach to sequencing problems (by Michael Held and Richard M. Karp, 1962)
        - Knapsack problems (by Hans Kellerer, Ulrich Pferschy, and David Pisinger, 2003)

- Week 5, 6: Still more on basic algorithm design techniques;
    - Lecture 5: Greedy technique
      Lec7.pdf ; demo of Dijkstra's algorithm ; demo of Interval Scheduling algorithm (by K. Wayne), Lec7-Heap.pdf ; Lec7-UnionFind.pdf ; DemoBinaryHeap.pdf (by Kevin Wayne) , DemoHeapify.pdf (by Kevin Wayne) ,
    - Lecture 2: Basic algorithm analysis techniques, worst-case, average-case, and amortized analysis;
      Lec2.pdf , demo of TableInsert (by C. Leiserson),
    - Reading material:
        - Chapter 2,15,16,7,33.4 of Introduction to Algorithms,
        - Chapter 5,4,6 of Algorithm design
        - a note by Sleator ,
        - A note on two problems in connexion with graphs (by E. W. Dijkstra, 1959)
        - Algorithm 97: Shortest path (by Robert W. Floyd, 1962)
        - Top-down analysis of path compression (by Raimund Seidel and Micha Sharir, 2005)
        - Set merging algorithms (by Hopcroft J. E., Ullman J. D., 1973)
        - Efficiency of equivalence algorithms (by M. Fischer, 1972)
        - Efficiency of a good but not linear set union algorithm (by R. Tarjan, 1975)
        - Worst-case analysis of set union algoritms (by R. Tarjan, 1984)
        - A theorem on Boolean matrics (by Stephen Warshall, 1962)
        - Efficient algorithms for shortest paths in sparse networks (by Donald B. Johnson, 1977)
        - Disjoint paths in networks (by J. W. Suurballe, 1974)
        - An interview with Edsger W. Dijkstra (Conducted by Philip L. Frana 2001)
        - On the shortest spanning subtree of a graph and the traveling salesman problem (by Joseph B. Kruskal Jr., 1955)
        - Shortest Connection Networks and Some Generalizations (by Robert. C. Prim, 1957)
        - A Mathematical Theory of Communication (by C. E. Shannon, 1948)
        - An interview with Claude Shannon (Conducted by Robert Price, 1982)
        - An interview with Robert M. Fano (Conducted by Arthur L. Norberg, 1989)
        - A Method for the Construction of Minimum-Redundancy Codes (by DAVID A. HUFFMAN, 1952)
        - Profile: David A. Huffman Encoding the â€œNeatnessâ€ of Ones and Zeroes (by Gary Stix, 1991)

- Binary Essence: Various aspects of data compression
- Algorithm 245, TreeSort 3 (by Robert M. Floyd, 1964)
- Discovery of Huffman Codes (by Inna Pivkina, 2010)
- Binomial Heap Script (by Sotirios Stergiopoulos, 2001)
- Fibonacci Heap Animation (by Jason Huang Hu and Wei Wang, 2003)
- What is a matroid? (by James Oxley, 2003)
- On the abstract properties of linear dependence (by Hassler Whitney, 1935)
- Non-separable and planar graphs (by Hassler Whitney, 1932)
- Matroids and greedy algorithms(by Jack Edmonds, 1971)
- A Data Structure for Manipulating Priority Queues (by Jean Vuillemin, 1978) ,
- Fibonacci heaps and their uses in improved network optimization algorithms (by M. Fredman and R. Tarjan, 1987) ,
- Robert Tarjan --- the art of the algorihtms (by Jamie Beckett, 2004) ,
- Amortized Analysis Explained (by Rebecca Fiebrink)

- Week 7, 8: Linear programming
  - Lecture 6: Linear programming: Simplex algorithm
    Lec8.pdf , an example of cycling in simplex algorithm (given by E. M. L. Beale, 1955), an example of Klee-Minty cube, a script to generate Klee-Minty cube (with noise), a script to generate Klee-Minty cube (without noise), the DIET problem (in.math format) ,

  - Reading material:
    - Chapter 29 of Introduction to Algorithms, Combinatorial optimization: algorithm and complexity.
    - The life and times of the father of linear programming (by Saul I. Gass) ,
    - An interview with George B. Dantzig: the farther of linear programming (Conducted by Watts, Griffis and McOuat, 1986) ,
    - Mathematical Methods of Organizing and Planning Production (by L. Kantorovich, 1939) ,
    - The First Algorithm for Linear Programming: An Analysis of Kantorovich's Method (by C. van de Panne and F. Rahnama, 1985) ,
    - CONCEPTS OF OPTIMALITY AND THEIR USES (Nobel memorial lecture, by T. Koopmans, 1975) ,
    - Mathematics in Economics: Achievements, Difficulties, Perspectives (Nobel memorial lecture, by L. Kantorovich, 1975) ,
    - The diet problem (by George B. Dantzig, 1990) ,
    - A primal-dual algorithm (by George B. Dantzig, L. R. Ford, D. R. Fulkerson 1956) ,
    - The cost of subsistence (by George Stigler, 1945) ,
    - Linear programming (by George B. Dantzig, 2002) ,
    - Linear programming and extensions PART I (by George B. Dantzig, 1963) ,
    - Linear programming and extensions PART II (by George B. Dantzig, 1963) ,
    - Ellipsoid Method (by Steffen Rebennack, 2008) ,
    - Lecture notes on the ellipsoid algorithm (by Michel Goemans, 2009) ,
    - The Ellipsoid Method: A Survey
    - Primal-Dual methods for linear programming (by Philip E. GILL, Walter MURRAY, Dulce B. PONCELEON and Michael A. SAUNDERS, 1994) ,
    - Interior point methods and linear programming (by Robert Robere, 2012) ,

        - ON PROJECTED NEWTON BARRIER METHODS FOR LINEAR PROGRAMMING AND AN EQUIVALENCE TO KARMARKAR'S PROJECTIVE METHOD (by Philip E. Gill, Walter MURRAY, Michael A. SAUNDERS, J.A. TOMLIN, Margaret H. WRIGHT, 1986) ,
        - A new polynomial-time algorithm for linear programming (by N. Karmarkar, 1984) ,
        - C++ implementation of Khachiya algorithm for the minimum enclosing (or covering) ellipsoid (by Bojan Nikolic) ,
        - In memoriam: Leonid Khachiyan ,
        - Klee-Minty example (by H. Greenberg) ,
        - Simplex examples ,
        - Smoothed complexity (by D. Spielman and S. Teng) ,
        - Smoothed Analysis of Algorithms: Why the Simplex Algorithm Usually Takes Polynomial Time (by D. Spielman and S. Teng, 2001) ,
        - GLPK (GNU Linear Programming Kit
        - A new polynomial-time algorithm for linear programming (by N. Karmarkar, 1984) ,
        - The interior-point revolution in optimization: history, recent developments, and lasting consequences (by Margaret H. Wright, 2004) ,
        - Why a pure primal Newton barrier step may be infeasible? (by Margaret H. Wright, 1995) ,
        - Numerical Optimization (by Nocedal, Jorge, Wright, S., 2006) ,
        - Solving Inequalities and Proving Farkasâ€™s Lemma Made Easy (by David Avis and Bohdan Kaluzny) ,

- Week 9, 10: Linear programming (cont'd)
    - Lecture 9: Linear programming: duality;
      Lec9.pdf , A gnuplot script to illustrate Lagrangian dual , Lec9-DIET.math , Lec9-DIET-Dual.math , Lec9-DIET-b1-2001.math , Lec9-DIET-b2-56.math , Lec9-DIET-b3-801.math
    - Reading material:
        - Chapter 29 of Introduction to Algorithms,
        - Combinatorial optimization: algorithm and complexity.
        - On the theory of games of strategy (by John von Neumann, 1928) ,
        - Non-Cooperative Games (by John Nash, 1951) ,
        - Zero-sum Two-person Games (by T. E. S. Raghavan, 1994) ,
        - KKT Examples (by Stanley B. Gershwin) ,
        - KKT conditions and applications (by Michel Baes) ,
        - Duality and KKT conditions (by S. Cui) ,
        - The Lagrangian Relaxation Method for Solving Integer Programming Problems (by Marshall Fisher, 2004) ,
        - Lagrange relaxation and KKT conditions () ,
        - Applied integer programming --- modelling and solution

- Week 11, 12: Network flow
    - Lecture 10: Network flow and its applications;
      Lec10.pdf , Network-flow applications , demo of Ford-Fulkerson algorithm , , demo of Edmonds-Karp algorithm , demo of Dinic algorithm , Irrational capacities might lead to endless iterations ,
    - Reading material:

- Chapter 26 of Introduction to Algorithms,
- Chapter 7 of Algorithm design, Combinatorial optimization: algorithm and complexity,
- Network-flow research history (by A. Schrijver) ,
- Maximal flow through a network (by L. R. Ford Jr. and D. R. Fulkerson, 1956) ,
- Algorithm for solution of a problem of maximum flow in a network with power estimation (by E. A. Dinic, 1970) ,
- Dinitz' Algorithm: The Original Version and Even's Version (by Yefim Dinitz, 2006) ,
- Finding disjoint paths in networks (by D. Sidhu, R. Nair, and S. Abdallsh, 1991) ,
- Theoretical Improvements in Algorithmic Efficiency for Network Flow Problems (by Jack Edmonds and Richard M. Karp, 1972) ,
- Network Flow Algorithms (Andrew V.Goldberg, Eva Tardos and Robert E. Tarjan, 1990) ,
- Maximum Matching and a Polyhedron With O,1-Vertices1 Jack Edmonds (by Jack Edmonds, 1964) ,
- Paths, Trees and Flowers ,
- Paths, Trees and Flowers (by Jack Edmonds, 1965) ,
- Faster scaling algorithms for general graph matching problems (by H. N. Gabow, R. Tarjan, 1991) ,
- A Scaling Algorithm for Maximum Weight Matching in Bipartite Graphs (by Ran Duan, Hsin-Hao Su, 2012) ,
- Efficient Algorithms for Finding Maximal Matching in Graphs (by Zvi Galil, 1983) ,
- Linear-Time Approximation for Maximum Weight Matching(by Ran Duan, Seth Pettie, 2014) ,
- Max-Product for Maximum Weight Matching: Convergence, Correctness, and LP Duality (by Mohsen Bayati, Devavrat Shah, and Mayank Sharma, 2008) ,
- A Primal Method for Minimal Cost Flows with Applications to the Assignment and Transportation Problems (by Morton Klein, 1967) ,
- Max flows in O(nm) time, or better (by James Orlin, 2012) ,
- Trees: A Mathematical Tool for All Seasons, including History of algorithms to find minimum cost spanning trees (by Joe Malkevitch)
- 50 Years of Integer Programming 1958-2008: From the Early Years to the State-of-the-Art, Springer, 2010 (Edited by M. Juenger, T. M. Kiebling, D. Naddef, G. L. Nemhauser, W. R. Pulleyblank, G. Reinelt, G. Rinaldi, and L. A. Wolsey)
- Finding minimum-cost circulations by successive approximation (by Andrew V. Goldberg, Robert E. Tarjan, 1987) ,
- What energy functions can be minimized via graph cuts (by Vladimir Kolmogorov, Ramin Zabih, 2004) ,
- Polyhedral Combinatorics and Combinatorial Optimization (by Alexander Schrijver) ,
- Two theorems in graph theory (by Clauder Berge, 1957) ,
- An $n^{5/2}$ Algorithm for Maximum Matchings in Bipartite Graphs (by J. Hopcroft, and R. Karp, 1973) ,
- On representatives of subsets (by P. Hall, 1935) ,
- A Primal Method for the Assignment and Transportation Problems (by M. L. Balinski and R. E. Gomory, 1964) ,
- The Hungarian Method for the Assignment Problem (by H. W. Kuhn, 1955) ,
- Variants of The Hungarian Method for Assignment Problems (by H. W. Kuhn, 1956) ,
- On the history of combinatorial optimization (by Alexander Schrijver, 2005) ,
- JenÂ¨o EgervÂ´ary: from the origins of the Hungarian algorithm to satellite

communication (by Silvano Martello, 2009) ,
    - On combinatorial properties of matrices (by Jeo Egervary, 1931, Translated by H. Kuhn) ,
    - Solutions of games by differential equations (by G. W. Brown, von Neumann, In: H. W. Kuhn and A. W. Tucker, Eds., Contributions to the Theory of Games I, Princeton University Press, Princeton, 1950) ,
    - A certain zero-sum two-person game equivalent to the optimal assignment problem (by von Neumann, In: H. W. Kuhn and A. W. Tucker, Eds., Contributions to the Theory of Games I, Princeton University Press, Princeton, 1950) ,
    - Algorithms for the assignment and transportation problems (by James Munkres, 1957) ,
    - A bibliography of graph matching (by Seth Pettie) ,
    - Differential gene expression analysis using coexpression and RNA-Seq data (by Tao Jiang et al. 2013) ,

- Week 13, 14: Problem intrinsic property: Hardness
  - Lecture 11: Problem hardness: Polynomial-time reduction;
    Lec3.pdf
  - Reading material:
    - Computer and intractability,
    - Chapter 8 of Algorithm design,
    - Chapter 34 of Introduction to Algorithms
    - Reducibility among combinatorial problems (by R. M. Karp, 1972) , slides
    - Molecular Computation of Solutions to Combinatorial Problems (by Leonard M. Adleman, 1994),
    - Computing with DNA (by Leonard M. Adleman, 1998),
    - Computer in a testtube (by Hendrik Jan Hoogeboom, 2010),
    - How to apply de Bruijn graphs to genome assemly (by Phillip E. C. Compeau, Pavel A. Pevzner, and Glenn Tesler, 2011),

- Week 14, 15: NP-Completeness
  - Lecture 12: NP-Hard problems: packing problem, covering problems, sequencing problem, partitioning, coloring, SAT, numbering problems, etc.
    Lec4.pdf , Turing machine demo (by Zhen Ji) , Turing machine demo (by Andrew Hodges, Turing machine (by K. Wayne) , Computability (by K. Wayne) , 2SAT is in P (by D. Moshko) ,
  - Reading material:
    - Computer and intractability,
    - Chapter 8 of Algorithm design,
    - Chapter 34 of Introduction to Algorithms
    - The complexity of theorem-proving procedures (by Stephen A. Cook, 1971)
    - The P versus NP problem (by Stephen Cook),
    - A compendium of NP optimization problems (Edited by Pierluigi Crescenzi, Viggo Kann, M. Halldorsson, M. Karpinski, and G. Woeinger)

- Week 16, 17: Solving hard problems: approximation and randomization
  - Lecture 11: Approximation algorithm: a brief introduction;
    Lec11.pdf , Parametric pruning algorithm for K-Center problem (by P. Potikas) , Lec11-SetCover-Primal.math , Lec11-SetCover-Dual.math , Lec11-MakeSpan.math ,
  - Reading material:

        - Chapter 35 of Introduction to Algorithms,
        - Chapter 11 of Algorithm design,
        - Approximation algorithm by V. Vazirani.
        - Branch and bound algorithms --- priciples and examples (by Jens Clausen, 1999)
        - Parameterized complexity and approximation algorithms (by Daniel Marx, 2005)
        - CS266 -- Parameterized Algorithms and Complexity (by Ryan Williams, 2013)
        - Invitation to parameterized algorithms (by Rolf Niedermeier, 2002)
    - Assignment 8
    - Hint of Assignment 8

- Week 18: Solving hard problems: approximation and randomization
    - Lecture 12: Randomized algorithm: a brief introduction;
      Lec12.pdf , Hashing (by K. Wayne)
    - Reading material:
        - Chapter 35 of Introduction to Algorithms,
        - Chapter 13 of Algorithm design,
        - Randomized Algorithm by R. Motwani and P. Raghavan.
          Randomized algorithms (by P. Raghavan) , Global Min-cuts in RNC, and other ramifications of a simple min-cut algorithm (by David R. Karger, 1992) ,

- Week 19: Solving hard problems: special cases and heuristics
    - Lecture 13: Extending limits of tractability;
      Lec13.pdf , TreePack: rapid side-chain prediction using tree-decomposition
    - Reading material:
        - Chapter 10 of Algorithm design,
        - lectures by D. P. Williamson.
        - Rapid side-chain prediction via tree decomposition (by Jinbo Xu, 2005) ,
        - Protein Threading using PROSPECT: Design and Evaluation (by Ying Xu and Dong Xu, 2000) ,
    - Lecture 14: Heuristics (local search strategy);
      Lec14 Local Search (by K. Wayne)
    - Reading material:
        - Chapter 11 of Algorithm design,
        - Combinatorial optimization: algorithm and complexity.

Powered by Loongson CPU  **. Loongson is a CPU developed at Institute of Computing Technology, Chinese Academy of Sciences.**