Understanding Algorithms

* Correctness

* Efficiency

  * Asymptotic complexity, O( ) notation

* Modelling

  * Graphs, data structures, decomposing the problem

* Techniques

  * Divide and conquer, greedy, dynamic programming

Expectations

* Background in programming

  * Any language (C, C++, Java)

* Basic data structures

  * Arrays, lists

Topics to be covered

* Asymptotic complexity

* Searching and sorting in arrays

  * Binary search, insertion sort, selection sort, merge sort, quick sort

* Graphs and graph algorithms

  * Representations, reachability, connectedness

  * Directed acyclic graphs

  * Shortest paths, Spanning trees

Topics to be covered

✳ Algorithmic design techniques

  ✳ Divide and conquer, Greedy algorithms, Dynamic programming

✳ Data structures

  ✳ Priority queues/heaps, Search trees, Union of disjoint sets (union-find)

✳ Miscellaneous topics

  ✳ Intractability, …

Tentative schedule

* Week 1: Motivation, asymptotic complexity

* Week 2: Searching and sorting

* Week 3: Graphs and basic graph algorithms

* Week 4: More graph algorithms, disjoint set

* Week 5: Divide and conquer, heaps

* Week 6: Search trees, greedy algorithms

* Week 7: Dynamic programming

* Week 8: Miscellaneous topics

January

4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	29	30	31

February

1	2	3	4	5	6	7
8	9	10	11	12	13	14
15	16	17	18	19	20	21
22	23	24	25	26	27	28

Evaluation

* Continuous evaluation

    * 8 Weekly quizzes

    * 6 programming assignments

* Certification exam

Requirement for successful course completion

* 60% in quizzes, certification exam

* Submit at least 5 of 6 assignments

    * At least 4 with nonzero marks

Textbooks

* **Algorithm Design**

  Jon Kleinberg and Eva Tardos

* **Algorithms**

  Sanjoy Dasgupta, Christos Papadimitriou and Umesh Vazirani