

NTUST, CSIE

Algorithms (CS3001-301, 必修), Spring 2021

Syllabus

Instructor: 臺科大資工系 鮑興國 Ph.D.
T4-505
Office Tel.: 2730-1065
pao@mail.ntust.edu.tw

Time & Place: Tue. 9:10 – 10:00, Wed. 10:20 – 12:10 (TR – 214)

Website: <http://moodle.ntust.edu.tw/>

Prerequisite: Data structures, some experience in high-level programming languages (e.g., C/C++, Java).

Textbook & Reference books:

Introduction to Algorithms (3rd Edition) by T.H.Cormen, C.E.Leiserson, R.L.Rivest, C.Stein, MIT Press (2009) (開發)

Grading:

| | | |
|----------|-------|-----|
| Midterm | | 30% |
| Final | | 30% |
| Homework | | 40% |

- 總共大約四至五個作業，其中可能含有一至兩個程式語言撰寫作業。
- 書面作業於上課前繳交，程式語言作業於期限當日午夜 12 時前繳交，遲交依遲交時間等比例扣分。
- 任何作業不允許抄襲，引用任何資料請註明出處。

Outline

(***bold** means a must)

- I. *Introduction (chapter 1, 3, 4)*
 - A. **Complexity and Growth of Functions**: asymptotic notation, common functions
 - B. **Divide and Conquer**
 - C. **Recurrences**: recurrence equations, master theorem
- II. *Data Structures (chapter 11, 12, 13, 18, 20, 21)*
 - A. **Trees**: binary search trees, red-black trees, B-trees, van Emde Boas trees
 - B. **Hashing**: direct-address tables, chaining, open addressing
 - C. **Heaps**: priority queues
 - D. **Sets**: data structures for disjoint sets
- III. *Basic and Advanced Algorithms (chapter 2, 6, 7, 8)*
 - A. **Basic Sorting Methods**: insertion sort, merge sort, selection sort, bubble sort
 - B. **Advanced Sorting Methods**: heapsort, quicksort, counting sort, radix sort
- IV. *Advanced Design and Analysis Techniques (chapter 15, 16)*
 - A. **Greedy Algorithms**: Huffman codes
 - B. **Dynamic Programming**: matrix-chain multiplication, longest common subsequence
- V. *Graph Algorithms (chapter 22 – 26)*
 - A. **Elementary Graph Algorithms**: breadth-first search, depth-first search, topological sort, strongly connected components
 - B. **Minimum Spanning Tree**: Prim's and Kruskal's algorithms
 - C. **Single-Source Shortest Paths**: Bellman-Ford algorithm, Dijkstra's algorithm
 - D. **All-pairs Shortest Paths**: Floyd-Warshall algorithm
 - E. **Maximum Flow**