School of Mathematics

Course 363 — Algorithms and complexity 2003–04
(Optional JS & SS Mathematics, SS Two-subjecct Moderatorship)

Lecturer: Colm Ó Dúnlaing

Requirements/prerequisites: There are no formal prerequisites. Students with little experience of data structures — lists, stacks, queues, trees, and so on — would not be advised to take this course. There will be some programming in Eiffel.

Duration: 21 weeks

Number of lectures per week: 3

Assessment: Homeworks counted.

End-of-year Examination: One 3-hour examination

Description: Sorting: mergesort, heapsort, quicksort, radix sort — worst-case and average analysis. Lower bound for sorting in comparison model. Searching: average-case analysis of binary search-trees, worst-case for red/black trees, amortised analysis of splay trees.

Amortised analysis of the UNION-FIND strategies. SPLIT-FIND and UNION-SPLIT-FIND (van Emde Boas trees). String matching: Knuth, Morris, Pratt and Boyer-Moore. Context-free grammars and Earley's algorithm. LR(1) parsers.

Digraphs and graphs. Depth-first search. Topological sort. Strong connectivity. Floyd-Warshall algorithm. Dijkstra's algorithm. Biconnectivity. Graph planarity. Network flows.

Hashing: uniform and linear. Perfect hashing.

Objectives: This course aims to cover a rather wide range of algorithms and analyse their efficiency.

June 14, 2004