For discussion by Toby Walsh
Feedback welcome Version 2

Analysis of Algorithms

1 Introduction

This proposed course is intended to be provide computing students with some of the core mathematical skills necessary to study and analyse computational problems, especially algorithms. Emphasis will be on teaching problem solving skills as opposed to mathematical theory.

2 Course outline

The course starts where the discrete mathematics course MATH 1081 and the algorithms courses COMP 3121 and 9101 stop. The necessary background materials from these courses will be briefly covered before moving on to the course goals of teaching students how to solve such problems in practice.

Summation

Maninpulation of sums. General methods: reduction to known form; induction; telescoping; perturbation; summation by parts.

Recurrences

Recurrences and sums. Linear homogenous recurrences. Linear nonhomogeneous recurrences. Generating functions.

Asymptotic analysis

Big O. Big Omega. Big Theta. Big O manipulation. Useful asymptotic approximations.

Complexity analysis

P. NP. Hardness, easiness and completeness. Basic problems like SAT, Hamiltonian circuit, and number partitioning. Proving NP-completeness: restriction; local replacement; gadget design. Dealing with NP-completeness. Approximation problems. Performance guarantees. Beyond NP.

0 Should not be distributed outwith APES and 4C.

Optimization

Linear programming. Simplex algorithm. Interior-point methods. Integer programming. Constraint programming.

3 Discussion

Students will be exposed to some computer tools like Maple and MATLAB to help them do such problem solving. In addition, there will be some guest lectures for other NICTA researchers on mathematical problem solving techniques used in their research.