PROPOSAL TO INTRODUCE A NEW COURSE

(formerly known as subject)

1.   COURSE DETAILS

1.1   Course ID   COMP4133

1.2   Course name - Long
Advanced Compiler Construction

1.3   Course name - Abbreviated
Advanced Compiler Construction

1.4   Course Authority   ext/email
Jingling Xue   jxue@cse.unsw.edu.au

1.5   Organisational Unit responsible for course

School: Computer Science and Engineering   **Faculty: Engineering**

Academic Group Code (Faculty): ENG
Academic Organisation Code (Owner): COMPSC

1.6   Justification of Proposal
This is a joint effort with Gabi Keller (COMP3161) and Manuel Chakravarty (COMP4132) in developing a Programming Languages and Compilers stream consisting of a set of coordinated courses, comparable to that offered in leading universities worldwide.
The existing COMP3131/9102 focuses on basic compiler techniques for implementing programming languages in a virtual machine (emphasis on compiler front ends). The proposed course will focus on compiling programming languages for efficient execution on modern (pipelined) processors (superscalar or VLIW) with a multi-level memory hierarchy. The course will cover (a) compiler techniques for designing and implementing compiler back ends, and (b) a number of research-oriented modern compiler techniques for these architectures. Programming assignments will provide experience with implementation issues and allow students to evaluate the impact of each compiler technique. The students are expected to gain a good understanding of fundamental principles in programming languages, develop further their programming and software engineering skills and be well prepared to undertake a thesis project in the areas of programming languages, compilers and computer architecture.
It is expected that this course will provide the foundation for a number of domain-specific compiler courses such as Compilers for High-Performance Computing (DMMs, SMMs, vectors and NOWs), Silicon Compilers (e.g., FPGAs) and Memory Management (e.g., Garbage collection and regions-based memory management techniques), which may be developed over the years for honours and PhD students.

1.7   Consultation Process
The Head and Associate Head of School and some members of the Teaching Committee.

1.8   Units of credit (UOC) Session/s offered   Hours Per Week
6   S1   3

1.9   Pre-requisites: a minimum 65% in COMP3131/9102
          Co-requisites: COMP3130 (Programming Language Paradigms) -- **RECOMMENDED**
          Exclusions:

1.10 Proposed Entry in the Faculty Handbook

Compiler Back Ends: (a) *program analysis* -- static single assignment form (SSA), control-flow analysis, data-flow analysis, abstract interpretation, dependence analysis, pointer analysis, type-based analysis; (b) *code optimisation*; (c*) code generation* -- register allocation, code selection and instruction scheduling. **Modern Compiler Techniques:** (a*) dynamic and staged compilation* -- profiling, specialisation, run-time code optimisation and generation; (b) *run-time support* -- memory management and garbage collection; (c) *compiler techniques for improving memory hierarchy performance* – control and data transformations, prefetching; (d) *compiler techniques for superscalar and VLIW architectures* -- predication, data speculation, control speculation, software pipelining.
The lecture materials will be complemented by two or three large programming assignments.

1.11 Is this course replacing an existing course?

NO X

1.12 Undergraduate / Postgraduate

1.13 Elective

1.14 Program stage
Usually taken in Stage IV (first offered in S1, 2003)

1.15 Program/s in which course is be available
BE (CE 3645; SE 3648; BioInf 3647), BSc (CS 3978), and the following postgraduate programs:
MEngSc 8685; McompSc 8680; MInfSc 8508; MSc 2665; ME 2765; PhD 1650

1.16 Proposed teaching methods and assessment practices
Examinable (formal) and assignments

1.17 Assessment grades to be used
Full range of grades (HD, DN, CR, PS, FL)

1.18 Mode of delivery

Internal   X

External

Other (specify)

1.19 Information Technology Requirements for students
Standard for Computer Science and Engineering

1.20 Textbooks

Set textbooks: none
Recommended References:
- Steven S. Muchnick, Advanced Compiler Design and Implementation, Morgan Kaufmann Publishers, 1997.
- Robert Morgan, Building an Optimizing Compiler, Digital Press, 1998.
- Alfred V. Aho, Ravi Sethi and Jeffrey D. Ullman, Compilers: Principles, Techniques and Tools, Addison-Wesley, 1986.
- Reinhard Wilhelm and Dieter Maurer, Compiler Design, Addison-Wesley, 1995.
- Michael Wofe, High-Performance Compilers for Parallel Computing, Addison-Wesley, 1996.
- Selected articles from conference proceedings of PLDI, POPL, PPoPP, OOPSLA, MICRO, ASPLOS, LCPC

1.21   Industrial experience component
N/A

2.   RESOURCE STATEMENT

2.1   Enrolments
Estimated or proposed enrolments for the next three years.

2003: 40

2004: 40

2005: 50

2.2   Resource Requirements

Staffing Requirements:

Hours per week
       Full-time Academic Staff: 4 hours/week (one semester only)
       Part-time Teaching Staff: N/A
       General Staff: Covered by standard support of the Computer Support Group

Field Costs: N/A

Studio/Laboratory Requirements: N/A

Materials Requirements: N/A

Equipment Costs: N/A

Computing Requirements: Use of existing facilities in the school

Library Requirements: Standard textbooks, journal articles and conference proceedings

Capital Funds Requirements: N/A

2.3   Servicing Implications:
N/A

2.4   Teaching Arrangements:

   YES

    NO   X

2.5   Alternative Delivery Arrangements:
N/A

2.6   Details of Tuition Fees:
Proposed fee:

$ for non-award enrolment (local)

$ for non-award enrolment (international)

$ for course which forms part of full fee-paying program (for local students)

$ for course which forms part of full fee-paying program (for international students)

3.   AUTHORISATION

3.1   Principal Librarian's Endorsement

I have examined the Library needs related to the above proposal and certify that existing Library holdings, staffing, services and accommodation are adequate / inadequate (delete one) to cover the demands that are inherent in it.

Appropriate arrangements for the use of digitised material to support this course have been made by the Course Authority with the Principal Librarian.

Further Comments:

Principal Librarian
  /   /2001

3.2   Head of School's Approval

I have examined the resource implications of the above proposal in regard to staff, space, materials, equipment, capital funds, and computing, and certify that the School can cover the demands that are inherent in it.

Further Comments:

Head of School
  /   /2001

3.3   Dean's Approval

I have examined the resource implications of the above proposal in regard to staff, space, materials, equipment, capital funds, and computing, and certify that:

3.3.1   (i)   the proposal involves no additional resources. (A statement from the Head of School explaining how this can be achieved must be provided); or

(ii)   the proposal involves additional resources and it is proposed to redeploy existing resources within the faculty. (A statement from the Head of School explaining how this will be achieved must be provided); or

(iii)   the proposal involves additional resources to be obtained as set out below; or

(iv)   the additional resources essential to bring the proposal into effect cannot be found within resources available to the faculty.

3.3.2   a fee will / will not (delete one) be charged for this course. If a fee is to be charged the Dean certifies as follows:

I have ensured that the Vice-Chancellor has been advised of the proposed fee arrangements, and note that approval of fee arrangements is needed before a new course can be implemented.

3.3.3   the proposal conforms to the University's commitment to Equal Opportunity in Education.

Statement from Head of School on Source of Additional Resources and/or Further Comments:

Dean
    /   /2001