PROPOSAL TO ESTABLISH A BACHELOR OF SOFTWARE ENGINEERING DEGREE

FACULTY OF ENGINEERING
UNIVERSITY OF VICTORIA

VERSION 6.0
NOVEMBER 8, 2002

1. Executive Summary

This document proposes a new software engineering program at the University of Victoria leading to the degree of *Bachelor of Software Engineering (BSENG)*. This degree program is to be offered jointly by two Departments in the Faculty of Engineering, the Department of Computer Science and the Department of Electrical and Computer Engineering. The cooperation of these departments attests to the true interdisciplinary nature of the software engineering field. The intent is to start the program in September 2003 by admitting students to both first and second year.

The requirements for this new program include 47 courses and mandatory work experience of 16 months. The courses, which are designed to be taken over eight terms, include 38 prescribed courses and nine elective courses. The proposed software engineering curriculum is innovative, attractive, and interdisciplinary.

The proposed software engineering degree program is designed to be accredited by both the *Canadian Engineering Accreditation Board (CEAB)* and the *Computer Science Accreditation Council (CSAC)*. Accreditation by both organizations befits the interdisciplinary nature of this field. Six Canadian software engineering programs have so far been accredited by CEAB [12, 13]. Three programs have been accredited by CSAC and thus meet the standards of the Canadian Information Processing Society [14]. Accreditation ensures that a program meets or exceeds the educational standards set by the accrediting bodies. CEAB, for example, specifies, in broad terms, the minimal program content required in mathematics, basic sciences, engineering science, engineering design, and complementary studies. As an ongoing process, accreditation will help ensure that the program is kept up to date and effective and that the curriculum meets the guidelines of the *Association of Professional Engineers and Geoscientists of British Columbia*.

The proposed new software engineering degree program integrates well with existing programs in the Faculty of Engineering and leverages available resources effectively. The proposed program structure follows the pattern of existing Bachelor of Engineering (BENG) programs at the University of Victoria. In particular, there is a common first year for engineering programs. This was adjusted to accommodate the proposed software engineering program. The new program also allows for transfers from University Colleges and could connect to the existing bridge from colleges into Third Year Engineering.

Full implementation of the proposed software engineering degree program will require a significant number of additional faculty members, including a Director of Software Engineering, in the Department of Computer Science and the Department of Electrical and Computer Engineering. Four faculty positions have already been filled in direct support of the existing Software Engineering Option in the Department of Computer Science and of the Software Engineering Specialization in the Department of Electrical and Computer Engineering. The additional positions will be created with funding from the *Doubling the Opportunity (DTO)* initiative [2].

The administrative structure of the BSENG program consists of six parts: the *Software Engineering Program Board*, the *Software Engineering Program Director*, the *BSENG*

Curriculum Committee, the *BSENG Work Experience Office*, the *Software Engineering Program Office*, and the Faculty of Engineering *Undergraduate Programs Coordination Committee*.

Many sources were consulted during the design of the curriculum of the proposed BSENG program. The IEEE/*ACM Computing Curricula Recommendation 2001* [7], including a draft of the *Software Engineering Volume* [8], was extensively consulted throughout the development of the proposed curriculum. The IEEE/ACM recommendations represent an intensive consultation of computing and software engineering practitioners and educators worldwide. A number of software engineering undergraduate degree programs, which have recently been started in Canada, the United States and other parts of the world, were also examined.

The degree program is built upon the current base of expertise in software engineering research and education in the Departments of Computer Science and Electrical and Computer Engineering. There have been software engineering research projects and courses at the University of Victoria since the late Seventies. The curriculum also draws on the experiences gained in offering a software engineering option and specialization in third and fourth year, since 1998. Over the years, the synergy between teaching and research in software engineering has contributed significantly to the graduate and undergraduate program. Thus, the proposed degree program is a logical next step in the development of software engineering as a discipline at the University of Victoria.

The proposed BSENG degree program meets Objective 12 of the *University of Victoria Strategic Plan* [1] since this high quality program constitutes a timely response to disciplinary and interdisciplinary developments and societal and student needs. This program is a good indication of the University's commitment and contribution to innovation in the information technology sector in British Columbia. Since software engineering is a high priority research area in the Faculty of Engineering, this program also meets Objective 16 by concentrating institutional support on one of the University of Victoria's focus areas.

Software engineering was identified as a Doubling the Opportunity program area for the University of Victoria in the March 11, 2002 letter to President Turpin from the *Deputy Minister of Advanced Education* [2]. Resources (i.e., staffing, funding, space, *etc*.) for the proposed Bachelor of Software Engineering degree are thus a major component of the University of Victoria plan for the DTO initiative. Positions and funding have been allocated. Space is being assigned in a new engineering building, currently at the design stage.

The structure of this document follows the proposal template specified in the *Guidelines and Procedures for the Submission of New Degree Programs in British Columbia*.

1.1 Background

For more than 30 years, the term "software engineer" has been used so broadly that virtually anyone who has written a line of computer software can call herself or himself a "software engineer." It has been common practice for software vendors to disclaim any responsibility for the consequences of defects in the software that they market. Those who write such software have rarely been held accountable. The result has been the

proliferation of software products that fail frequently and incur significant costs to their users. For software used in safety-critical applications, such as medical, avionics, and automobile control systems, the risks of failure are especially high.

Software engineering is the application of computer science and engineering principles and practices to the design, construction, testing, maintenance, and evolution of software. While the term itself was coined in the late Sixties, it is only recently that software engineering has emerged as an undergraduate discipline in its own right. It is essential to the economic growth and development in the Province of British Columbia that students have the opportunity to study this increasingly critical area and obtain *software engineering degrees* [10].

Today, the documentation, design, development, evolution, and verification of computer software are integral parts of engineering solutions. The focus of the proposed software engineering degree program will be both on the application of engineering methods to software, and on the development of engineering software applications. There will be special emphasis on three themes: (1) the analysis and testing of software for correctness, performance, and usability; (2) the design of software to reduce the costs of maintenance and evolution; and (3) the design of software as part of larger, embedded, and real-time systems.

1.2 Objectives

The major objectives of the proposed program are:

- to satisfy student demand and industry demand for software engineering education in British Columbia;
- to give students a thorough and deep understanding of the specification, development, implementation, testing, maintenance, and evolution of software systems ranging from embedded systems to large information systems;
- to create an innovative program to address the truly interdisciplinary nature of software engineering effectively;
- to build upon the substantial software engineering research and education expertise already present at the University of Victoria; and
- to deploy the quality and strengths of University of Victoria Engineering Faculty into software engineering education.

1.3 Outcomes

The principal outcome of the proposed program will be graduates with a breadth and depth of knowledge of software engineering that combines theory and practice of Computer Science and Engineering. Graduates will also have strong communication skills and be proficient in engineering design, particularly as it applies to software development and software systems. Software engineering graduates will also start companies contributing to the economic growth of British Columbia.

The knowledge and skills that graduates will be expected to acquire during the BSENG program include:

- an understanding of all aspects of software development and the software development process from the early design stages to long-term software maintenance and evolution;
- the ability to construct and evaluate software in the context of physical systems and real-world applications;
- the ability to apply software and engineering design principles to software development and system design including trade-off analyses;
- an understanding of software quality criteria and assurance and the ability to assess the quality of a software system;
- the ability to plan and manage large software projects;
- the ability to work independently and collaboratively;
- an understanding of engineering economics and entrepreneurship in software practice;
- the ability to understand the underlying principles on which physical systems and real-world applications are built on;
- the ability to integrate and participate in the design process of these systems and applications;
- the capability to communicate effectively both orally and in writing; and
- a breadth of background, knowledge, and skills in software engineering, as well as related areas of engineering, computer science, mathematics and complementary studies, that will provide students with a base for life-long learning.

2. Letter of Intent

The *Letter of Intent to Offer a Bachelor of Software Engineering* dated January 16, 2002 is listed as Appendix A of this document. Appendix B lists the comments received from other institutions regarding the Letter of Intent.

3. Curriculum

3.1 Knowledge and Skills

Entrepreneurship will become a core skill which all our young people will need to exploit the opportunities emerging from science and technology, culture and communications.

Tony Blair, British Prime Minister

A number of software engineering undergraduate degree programs have recently been started in Canada, the United States and other parts of the world. The proposed program is the first full undergraduate program in Software Engineering to be offered in British Columbia. In that respect, it is the first opportunity for B.C. students to pursue the knowledge and skills required for a career as a software engineering professional in a focussed program of study leading to a *Bachelor of Software Engineering* degree. The

curriculum is also founded on the experiences gained in offering a software engineering option and specialization in third and fourth year since 1998 at the University of Victoria.

The curriculum has been designed to give students a thorough and deep understanding of the specification, development, implementation, testing, maintenance, and evolution of software systems ranging from embedded systems to large information systems. As a balanced engineering curriculum, the courses ensure that students obtain a foundation in mathematics and basic sciences, a broad preparation in engineering design and engineering sciences, and an exposure to non-technical subjects in the form of complementary studies courses. The core of the curriculum consists of software engineering, computer science, electrical engineering, and computer engineering courses.

Since many students are undecided about the discipline of study when they first come to university, the first year in Engineering at the University of Victoria is common to all programs. The first year courses emphasize fundamental skills, such as computer programming, engineering design, basic science, calculus and English. These courses also provide the broad background necessary to allow students to choose from the broad range of engineering and basic science electives. Student advising is a critically important part of Engineering First Year. The primary goals of the two courses on electrical and mechanical systems are to develop engineering design skills and to expose students to a variety of systems.

The second and third year courses develop engineering skills and fundamental knowledge, which are essential for a software engineering degree, including software engineering, computer science, computer engineering, electrical engineering, basic science, business, entrepreneurship, technical writing, and elective complementary studies.

Fourth year continues this skill and knowledge development with an additional four prescribed courses. An engineering curriculum must culminate in a significant design experience which is based on the knowledge and skills acquired in earlier course work and which, preferably, gives students an exposure to the concepts of teamwork. The required capstone project is intended for this purpose. The fourth year of the proposed software engineering degree program also includes five engineering electives. These courses allow the students to specialize and pursue particular interests in greater depth. The final course in this software engineering degree program introduces students to the legal, social, and professional issues that arise in software engineering practice.

The selection of skills and knowledge to be taught in this program was guided by the IEEE/ACM Computing Curricula Recommendation 2001, existing software engineering degree programs, accreditation requirements, and the expertise of the faculty in the Department of Computer Science and the Department of Electrical and Computer Engineering. The result is a balanced state-of-the-art curriculum.

Work experience is a mandatory component of the degree program. This is critical to providing students the opportunity to practise and apply knowledge and skills acquired throughout this software engineering program.

In summary, the knowledge and skills that graduates will be expected to acquire during the BSENG program include but are not limited to:

- an understanding of all aspects of software development and the software development process from the early design stages to long-term software maintenance and evolution;
- the ability to construct and evaluate software in the context of physical systems and real-world applications;
- the ability to apply engineering design principles to software development including trade-off analyses;
- an understanding of software quality criteria and assurance and the ability to assess the quality of a software system;
- the ability to plan and manage large software projects;
- the ability to work independently and collaboratively;
- an understanding of engineering economics and entrepreneurship in software practice;
- the ability to understand the underlying principles on which physical systems and real-world applications are built on;
- the ability to integrate and participate in the design process of these systems and applications;
- the capability to communicate effectively both orally and in writing; and
- a breadth of knowledge and skills in software engineering, as well as related areas of engineering, computer science, mathematics and complementary studies, that will provide students with a base for life-long learning.

3.2 Course Requirements

The requirements for this new BSENG program include 47 courses and mandatory work experience of 16 months. The courses, which are designed to be taken over eight terms, include 38 prescribed courses and nine elective courses.

The proposed program structure follows the pattern of the existing BENG programs at the University of Victoria. Most of the courses are prescribed in any engineering degree program. The prerequisite structure is well defined. A balanced distribution of basic science, mathematics, engineering science, engineering design, and complementary studies is required. An engineering program must culminate in a significant design experience and, hence, the capstone project in fourth year. This project is usually one term, but can also be extended to two terms. Moreover, two different projects can be undertaken for full credit in the two fourth-year terms. Over half of the required core courses have a laboratory component, which involves substantial projects.

The main part of the software engineering core is a sequence of eight courses (labeled SE1-SE8 below). These courses are designed to be taken in order to provide a coordinated development of the student's software engineering expertise. All courses in this series will include a laboratory component.

The program has been designed to take advantage of existing courses and expertise in the Faculty of Engineering. New courses are introduced only as necessary for the academic structure of the proposed program, but many existing courses will be redesigned to

accommodate this new degree program. The tables below show that the program specifies twelve new courses (i.e., WE, SYSDYN, SAS, BUS1, BUS2, CTRL, SEC, IKM, CON, CSCW, WMC, and CAP). Eight of these are part of the required core and four courses are engineering electives. Courses SE3 to SE8 are new courses, but replace existing SENG courses. Existing courses which require adjustments include SE1, SE2, MECHSYS, ELECSYS, CAS, DD, OSDC, ARCH, CBSE, and SOCIAL. To accommodate the additional BSENG students, new sections will have to be created for selected existing courses (e.g., SE1 and SE2).

The BSENG curriculum development process is described in the *Software Engineering Degree Program: Executive Summary* [5]. The BSENG curriculum is detailed in two course description documents, a calendar entry document, and the tables below. Appendix D lists short, calendar-style course descriptions for all courses of the proposed software engineering degree program [3]. For detailed course outlines please refer to the 75-page document *Software Engineering Degree Program: Detailed Course Descriptions* [4]. For an overview of the proposed First Year of Operations please consult the *Bachelor of Software Engineering Calendar Entry* [15].
In the tables below, the first two columns list the unique identifier and the course subject, respectively. The last two columns list the status of a course and a reference to an existing or related course, where

E – denotes an existing course requiring no modification;

M – denotes an existing course likely requiring modification over time; and

N – denotes a new course.

First Year

The existing BENG programs at the University of Victoria have a common first year. The first year in the Computer Science programs is quite different. The first year of the BSENG shown in the following two tables is similar to the existing BENG first year, but can also serve as preparation for computer science degrees. This approach ensures maximal flexibility for students to change programs during or even after first year with minimal additional course work. SE1 and SE2 are new sections of two existing computer science courses (i.e., CSC 110 and CSC 115/CSC160).

TERM 1A	SUBJECT		EXISTING
SE1	Fundamentals of Programming	M	CSC 110
MECHSYS	Mechanical Systems	M	MECH 141
LA	Linear Algebra	E	MATH 133
CALC1	Calculus I	E	MATH 100
PHYS	Physics	E	PHYS 125

TERM 1B	SUBJECT		EXISTING
SE2	Object Oriented Design and Methodology	M	CSC 115 or CSC160
ELECSYS	Electrical Systems	M	ELEC 199
WE	Web Engineering	N	None
CALC2	Calculus II	E	MATH 101
CHEM	Chemistry	E	CHEM 150
ENGL	English	E	ENGL 115

Second Year

The discrete mathematics components in second year are essential for several software engineering courses. Ideally these courses would be taught in first year, but, in order to achieve a common first year, these courses are assigned to second year.

TERM 2A	SUBJECT		EXISTING
SE3	Software Development and Architecture	N	Note 1
CAS	Computer Architecture and Assembly Language Prog.	M	CSC 230
PS	Introduction to Probability and Statistics	E	STAT 260
SYSDYN	System Dynamics	N	None
DS1	Discrete Mathematics I	E	MATH 122
TECHWRT	Technical Writing	E	ENGR 240

TERM 2B	SUBJECT		EXISTING
SE4	Systems Programming and Middleware	N	Note 1
ALG1	Algorithms and Data Structures	E	CSC 225
HCI	Human Computer Interfaces	E	SENG 310
SAS	Signals and Systems	N	ELEC 260/310
DS2	Discrete Mathematics II	E	MATH 222
BUS1	Economics and Entrepreneurship	M	ENGR 280

Note 1: Courses *SE3-SE8* are new but in fact in combination replace existing SENG 265, 330, 365, 420, 440 and 465.

Third Year

The third year of the program permits two basic science electives and one complementary study elective to allow each student to broaden his or her knowledge. The two business components in second and third year are essential for software engineering practice.

TERM 3A	SUBJECT		EXISTING
SE5	Requirements Engineering and Formal Specification	N	Note 1
DD	Digital Design	M	CSC 355 or CENG 355
OSDC	Introduction to Operating Systems	M	CSC 360
CTRL	Control Systems	N	ELEC 360 or MECH 435
BSC5	Basic Science Elective	E	various
BUS2	Planning and Management	N	None

TERM 3B	SUBJECT		EXISTING
SE6	Software Evolution	N	Note 1
AFL	Foundations of Computer Science	E	CSC 320
DB	Databases	E	CSC 370
SEC	Security Engineering	N	Note 2
BSC6	Basic Science Elective	E	various
CST5	Complementary Studies Elective	E	various

Note 2: The new course *SEC* replaces the existing course *SENG 424: Reliability*. Due to staff shortages, SENG 424 has never been offered in the existing programs.

Fourth Year

The fourth year of the program permits five engineering electives and one free elective to allow each student to focus his or her studies in the area of most interest.

TERM 4A	SUBJECT		EXISTING
SE7	Embedded Systems	N	Note 1
NET	Computer Networks	E	CSC 450 or CENG 460
Elective	A course taken from the engineering electives listed below		Tables A and B
Elective	A course taken from the engineering electives listed below		Tables A and B
Elective	A course taken from the engineering electives listed below		Tables A and B
Free Elect.	A free elective taken from any Faculty		Any course

TERM 4B	SUBJECT		EXISTING
SE8	Software Quality Engineering	N	Note 1
CAP	Capstone Design Project	N	SENG 499
RT	Real-Time Systems	E	CSC 460 or CENG 455
Elective	A course taken from the engineering electives listed below		Tables A and B
Elective	A course taken from the engineering electives listed below		Tables A and B
SOCIAL	Social and Professional Issues	N	ENGR 498 or SENG 400

Engineering Electives

Students can choose courses from the following two tables of core engineering electives for the five slots of fourth year engineering electives. To satisfy accreditation requirements, at least three of these courses must come from Table A. The remaining two courses may be chosen from Table A or Table B. However, with permission of the BSENG Program Director, most other courses with prefixes CSC, CENG, and ELEC may also be used to satisfy this elective requirement. The sixth or free elective course may be selected at any level and from any Faculty, including the Faculty of Engineering.

Engineering Electives Table A

YEAR 4	SUBJECT		EXISTING
CAP	Capstone Design Project (to allow two-term projects)	E	SENG 499
FTC	Fault-tolerant Computing	E	CSC 454
ALG2	Algorithms and Data Structures 2	E	CSC 326
ALG3	Analysis of Algorithms	E	CSC 425
CG	Computer Graphics	E	CSC 405
MMS	Multimedia Systems	E	SENG 410 or CSC 461
IKM	Information and Knowledge Management	N	None
NC	Network-centric Computing	M	SENG 450
DC	Distributed Computing	M	SENG 462 or CSC 462
MICRO	Microprocessor Systems	M	CENG 355
CSA	Computer Systems and Architecture	E	CENG 450
AI	Artificial Intelligence	E	CENG 420
PATREC	Pattern Recognition	E	ELEC 485
ROBOT	Robotics	E	ELEC 426 or MECH 430
ERGO	Ergonomics	E	SENG 412

Engineering Electives Table B

YEAR 4	SUBJECT		EXISTING
ARCH	Software Architecture	M	SENG 422
CBSE	Component Based Software Engineering	N	Topics course
CC	Compiler Construction	E	CSC 435
PL	Programming Languages	E	CSC 330
CON	Concurrency	N	None
CSCW	Computer Supported Collaborative Work	N	None
DSP	Digital Signal Processing	M	ELEC 310
COM	Communications	M	ELEC 350
MCOM	Mobile Communications	E	ELEC 456
WMC	Wireless and Mobile Computing	N	None
NUM	Numerical Methods	E	CSC 340
NA1	Numerical Analysis I	E	CSC 349A
NA2	Numerical Analysis II	E	CSC 349B
ORLP	Operations Research: Linear Programming	E	CSC 448A
ORSIM	Operations Research: Simulation	E	CSC 448B
MIN	Data Mining	N	None
SP	Software Process	E	SENG 470
AOOP	Advanced Object Oriented Design	E	SENG 430
MSD	Management of Software Development	E	SENG 472

The following table summarizes the components and the contributions of the key disciplines to the proposed software engineering degree program.

BSENG COMPONENT	NUMBER OF COURSES
Software Engineering	12
Engineering	7
Computer Science	6
Mathematics	6
Basic Science	4
Complementary Studies Electives	7
Engineering Electives	5
Total	47

3.3 Program Structure and Class Size

The program itself will have no additional specialties, minors or majors. Students will however still be able to pursue the *Software Engineering Option* within a Computer Science degree or a *Software Engineering Specialization* within a Computer Engineering degree.

The program size is expected to be 75 students per year. As for other programs in the Faculty, class size will vary depending on the level and whether the course is shared with other programs. Class size typically ranges from 150 – 175 in first year to 15 – 60 in fourth year electives. The student to faculty and student to teaching assistant and/or laboratory instructor ratios will be similar to existing programs.

3.4 Research Expectations

There have been software engineering research projects and courses at the University of Victoria since the late 1970's. The synergy between teaching and research in software engineering at the University of Victoria is exemplary. Undergraduate students participate in software engineering research projects and often continue as graduate students. Examples are Margaret-Anne Storey and Ken Wong, who started as undergraduates in computer science, became—as undergraduates—involved in software engineering research projects, and obtained their M.Sc. and Ph.D. degrees. Today they are software engineering professors at the Universities of Victoria and Alberta, respectively.

As an undergraduate program, the BSENG degree program will be primarily course based. However, many courses will include significant projects and all students will be required to complete a capstone design project. These projects will provide an opportunity for faculty to share their research expertise with students and, in many instances, for students to undertake research-related activities.

4. Learning Methodologies

4.1 Learning Environment

The learning environment will be similar to existing Electrical and Computer Engineering and Computer Science programs as the proposed BSENG program makes substantial use of existing courses. The learning environment is multifaceted and consists of classroom instruction, laboratory tutorials and projects, the capstone design project, and the mandatory work experience.

Advising is an important component throughout the proposed degree program. The Software Engineering Program Director ensures appropriate advising for BSENG students. The BSENG student advising team includes:

- The Software Engineering Program Director
- One faculty member from the Department of Computer Science
- One faculty member from the Department of Electrical and Computer Engineering
- The BSENG Program Office
- The BENG Programs Office

The BENG Programs Office advises first year students, who are in the common engineering first year. The BSENG Program Office is expected to advise mostly 2nd, 3rd, and 4th year BSENG students.

The Director also handles concerns and appeals regarding the BSENG program. Student concerns regarding a particular course shall, in the first instance, be handled by the course instructor, second by the Software Engineering Program Director, third by the appropriate Department Chair, and fourth by the Associate Dean or the Dean of Engineering.

4.2 Learning Methods

Experiential Learning—Work experience is mandatory for the existing BENG and optional for the existing Computer Science degree programs. The proposed BSENG has a mandatory work experience. The minimum total length of the work experience is 16 months. The requirement can be completed with the acquisition of an aggregate of at least 16 months of full time work periods. The shortest period of full time work recognized towards the aggregate is one month, but periods could range from 1 to 16 months, thus giving students ample flexibility. Under special circumstances students may be able to complete a portion of the work experience utilizing periods of half time work coupled with half time study. However, accumulations of "casual" work hours may not be applied to the completion of the work experience. Approval of special arrangements will be at the discretion of the Director of the BSENG Program and the Executive Director of Co-operative Education or their designates. Challenges or transfer of credit for up to eight months of the work experience are also permitted in the case of extensive relevant work experience *prior* to entering the program or appropriate work terms completed at a certified post-secondary institution. Regardless of the extent of prior work

experience, at least eight months of the work experience must be completed while the student is registered at the University of Victoria.

In addition to the above work experience, most courses include laboratory components, assignment, and projects emphasizing experiential learning. All students are required to complete a four-month capstone design project that can be extended to eight months.

Employability Skills—The proposed BSENG program covers a broad range of materials from theory to practice. The emphasis is on providing students with the necessary skills for careers as software engineers and the background for life-long learning required to succeed in the fast moving world of engineering and technology. The emphasis is not directly on skills for specific jobs, but clearly much of the course materials combined with the mandatory work experience will make graduates of this program eminently employable [10].

Lectures, Labs, and Tutorials—Courses will be primarily delivered in the on-site lecture mode. Labs and tutorials will form part of the course when appropriate. Labs can either be 'structured' (i.e., a sequence of experiments is executed with prescribed methods in a specific time slot) or 'non-structured' (i.e., substantial software development assignments and projects are carried out by the students using facilities provided by the University, but not in a specific time slot or under specific time constraints). However, for both lab types, progress and success is measured against specified performance requirements.

5. Faculty

Full implementation of the proposed software engineering degree program will require a significant number of additional faculty members, including a Director of Software Engineering, in the Department of Computer Science and the Department of Electrical and Computer Engineering. Four faculty positions have already been filled in direct support of the existing Software Engineering Option in the Department of Computer Science and of the Software Engineering Specialization in the Department of Electrical and Computer Engineering. The additional positions will be created with funding from the *Doubling the Opportunity (DTO)* initiative [2].

Anticipating ten filled positions in support of the software engineering program by July 1, 2003, there will be sufficient Faculty resources to start first and second year of the proposed degree program in September 2003. Further positions will be allocated in support of the software engineering program under subsequent years of DTO funding to start the third and fourth year of the degree program.

In addition to the positions above, several other faculty members are involved in software engineering research and education. A number of other members of the Faculty of Engineering have interests related to software engineering or teach courses that will be in direct support of the program.

Research and teaching profiles for the faculty members noted above can be accessed via the Department home pages (i.e., www.csc.uvic.ca and www.ece.uvic.ca).

6. Program Consultations and Evaluation

The program is built upon the current base of expertise in software engineering research and education in the Departments of Computer Science and Electrical and Computer Engineering at the University of Victoria. There are at present six faculty members whose research is specifically in the area of software engineering. A software engineering program option and specialization have been offered since 1998 in the Faculty of Engineering. The proposed program is thus a logical next step in the development of software engineering as a discipline at the University of Victoria.

6.1 Consultation

Many sources were consulted during the design of the curriculum of the proposed BSENG program. The IEEE/ACM[1] Computing Curricula Recommendation 2001 [7], including a draft of the Software Engineering Volume [8], was extensively consulted throughout the development of the proposed curriculum. The course descriptions of this IEEE/ACM recommendation were a critical source for the design of the BSENG program. The IEEE/ACM recommendations represent a fully intensive consultation of computing and software engineering practitioners and educators worldwide.

A number of software engineering undergraduate degree programs, which have recently been started in Canada, the United States and other parts of the world, were also consulted during the development of the proposed BSENG curriculum. The following table lists some of the specific software engineering programs which were consulted. University of Waterloo's *Bachelor of Software Engineering (BSE)* [11] is probably the program most closely related to the proposed University of Victoria BSENG program. In particular, the Waterloo program is also jointly offered by the School of Computer Science and the Department of Electrical and Computer Engineering, but the two units are in different faculties.

INSTITUION	WEB SITES
Butler University	http://www.butler.edu/cs/
California Polytechnic Institute	http://ceng-web.calpoly.edu/computer_sci.html
Carleton University	http://www.carleton.ca/cu/ed4life/brochures/soft_eng.html
Concordia University	http://www.cs.concordia.ca/programs/ugrad/soen/soen.html
Drexel University	http://www.ece.drexel.edu/ECE/software.html
Lakehead University	http://www.lakeheadu.ca/~engwww/soft/tech_sched.html
McMaster University	http://www.cas.mcmaster.ca/cas/undergraduate/SEprogrammes.html http://www.eng.mcmaster.ca/Engineering1/interested/eng1.html
Milwaukee School of Engineering	http://www.msoe.edu/eecs/se/
Monmouth University	http://www.monmouth.edu/academics/ugprograms/seug.asp

[1] The Institute of Electrical and Electronics Engineers (IEEE) and the Association of Computing Machinery (ACM) are the principal worldwide organizations for computing professionals.

Rochester Institute of Technology	http://www.se.rit.edu/degrees.php
University of Calgary	http://www.eng.ucalgary.ca/students/programs/student_software.htm
University of Ottawa	http://www.uottawa.ca/academic/info/regist/crs/0203/gngEN/gng-eng11.htm#4
University of Texas at Dallas	http://www.utdallas.edu/dept/eecs/se.html
University of Waterloo	http://www.softeng.uwaterloo.ca/
University of Wisconsin	http://www.uwplatt.edu/~se/
University of Western Ontario	http://www.engga.uwo.ca/electrical/default.htm

6.2 Evaluation

The program has been subject to the normal review process at the University of Victoria and was approved for submission by the *Senate Committee on Planning*.

***Other British Columbia Institutions*—**Other British Columbia Institutions had the opportunity to comment upon the *Letter of Intent* in the normal manner through the *Degree Program Review Committee (DPRC)*. Comments received from DPRC are listed in Appendix B.

Institutions Outside of British Columbia—As noted, the programs at 16 universities, including eight in Canada, were reviewed during curriculum development. Software Engineering curriculum and accreditation have been topics of ongoing discussion at the national meetings of the Deans of Engineering and the Chairs of Computer Science for several years. University of Victoria representatives were active participants in these discussions.

External academic consultants—In May 2002, the Chair of the BSENG Curriculum Committee presented the proposed BSENG degree program to several leading academics at the *IEEE/ACM International Conference on Software Engineering (ICSE 2002)*.

6.3 Accreditation

***Canadian Engineering Accreditation Board (CEAB) and the Computer Science Accreditation Council (CSAC)*—**Accreditation is designed to ensure that a program meets or exceeds the educational standards set by the accrediting bodies. CEAB for example specifies, in broad terms, the minimal program content required in mathematics, basic sciences, engineering science, engineering design, and complementary studies. The CSAC guidelines are similar.

The program has been constructed to meet both the CEAB and CSAC guidelines. We will apply for accreditation as soon as permitted (i.e., the year the first group of students graduate from the program). We will also discuss the program with the accreditation bodies and take advantage of pre-reviews that might be available.

Our assessment of the proposed program relative to CEAB course content criteria is shown in the following table. One AU (Accreditation Unit) corresponds to one lecture hour and 0.5 AU for each lab or tutorial hour.

BSENG SUBJECT AREA	Minimum AUs	Estimated AUs
Mathematics	195	402
Basic Sciences	225	300
Complementary Studies	225	278
Engineering Sciences	225	470
Engineering Design	225	446
Engineering Sciences and Design together	900	916
Total AUs in BSENG Program	1,800	1,896

Other aspects of the curriculum content criteria relate to laboratory experience, the roles of professional engineers in society, and approaches to independent learning. We anticipate these criteria will be readily satisfied by the proposed BSENG program.

CEAB and CSAC accreditation considers such issues as program leadership; the quality of the students, staff and faculty; laboratories; facilities including computing resources; library resources; and program governance. These have been considered in the program resource and management structure including the creation of the position of Director of Software Engineering [6].

As an ongoing process, accreditation will help ensure that the program is kept up to date and effective and that the curriculum content meets the guidelines of the *Association of Professional Engineers and Geoscientists of British Columbia*.

Over the last two years, six Canadian software engineering programs (i.e., Calgary, Concordia, Lakehead, McMaster, Ottawa, and Western Ontario) were accredited by CEAB [12, 13]. CEAB expects to evaluate ten other software engineering programs for possible accreditation within the next three years [13]. Three programs (i.e., Calgary, Saskatchewan, and Western Ontario) have been accredited by CSAC and thus meet the standards of the Canadian Information Processing Society [14].

6.4 Letters of Support

Appendix C lists letters in support of the BSENG degree program at the University of Victoria from leaders of the information technology sector in British Columbia and elsewhere.

7. Admission and Transfer

The BSENG degree program at the University of Victoria is designed to be a direct entry undergraduate program. Because the first year is the same for all engineering programs, students can complete first year and then choose among the software engineering, computer science, electrical engineering, computer engineering, and mechanical engineering programs. Students must declare this program as a major no later than the end of their second year of study.

The proposed software engineering program also allows for transfers from University Colleges and could connect to the existing bridge from colleges into Third Year Engineering. The Faculty of Engineering has well established transfer processes.

Work experience challenge based on appropriate prior employment will be permitted. Course challenge based on prior learning will not be permitted, but transfer credit for equivalent courses from other post-secondary institutions will be. This is consistent with current practice in the Faculty of Engineering.

8. Administrative Structure

The administrative structure document [6] outlines the mechanisms that implement the principles of equal sharing and joint management by the Department of Computer Science and the Department of Electrical and Computer Engineering in the Faculty of Engineering. It also secures the unique identity of the new program while ensuring that the operation and authority of the two home departments and their administrative structures remain intact.

The implementation of these principles is accomplished through the introduction of the following administrative structures: the *Software Engineering Program Board*, the *Software Engineering Program Director*, the *BSENG Curriculum Committee*, the *BSENG Work Experience Office*, the *Software Engineering Program Office*, and the *Faculty of Engineering Undergraduate Programs Coordination Committee*.

Introducing a new interdisciplinary program, such as the BSENG program, is a balancing act. All programs in the Faculty of Engineering are important. The BSENG program should not grow to the detriment of other programs. A key overall objective is to offer high-quality programs across the entire Faculty of Engineering.

In summary, the administrative structure provides effective mechanisms for faculty and staff to deliver high-quality software engineering programs. Thus, students in the proposed University of Victoria Bachelor of Software Engineering program can concentrate on acquiring knowledge and skills through the balanced state-of-the-art BSENG curriculum.

References

[1] Planning and Priorities Committee. A Vision of the Future: A Strategic Plan for the University of Victoria, 30 pages, February 2002.
http://web.uvic.ca/univsec/Strategic.pdf

[2] Letter from the Deputy Minister of Advanced Education to the President of the University of Victoria identifying Software Engineering as a Doubling the Opportunity Program Area, March 2002.
http://www.aved.gov.bc.ca/news/nr2002/nrae006-02.html

[3] Bachelor of Software Engineering (BSENG): Calendar Course Descriptions, 13 pages, Faculty of Engineering, University of Victoria, November 2002.
http://www.csr.uvic.ca/~hausi/sedp/BSENG-cal-course-desc-V6.0.pdf

[4] Bachelor of Software Engineering (BSENG): Detailed Course Descriptions, 76 pages, Faculty of Engineering, University of Victoria, August 2002.
http://www.csr.uvic.ca/~hausi/sedp/BSENG-course-desc-V4.5.pdf

[5] Bachelor of Software Engineering (BSENG): Curriculum Executive Summary, 8 pages, Faculty of Engineering, University of Victoria, August 2002. http://www.csr.uvic.ca/~hausi/sedp/BSENG-cur-exec-sum-V4.5.pdf

[6] Software Engineering Administrative Structure, Faculty of Engineering, 9 pages, University of Victoria, August 2002. http://www.csr.uvic.ca/~hausi/sedp/BSENG-admin-V4.5.pdf

[7] IEEE/ACM Computing Curricula 2001 (Steelman Draft), August 2001. http://www.computer.org/education/cc2001/steelman/cc2001/index.htm

[8] IEEE/ACM Computing Curricula: Software Engineering Volume (First Draft), August 2002. http://sites.computer.org/ccse

[9] Guide to the Software Engineering Body of Knowledge, April 2001. http://www.swebok.org/

[10] Wolfgang Strigel. The Canadian Software Industry, *IEEE Software*, July/August 2002.

[11] University of Waterloo Bachelor of Software Engineering (BSE). http://www.softeng.uwaterloo.ca/

[12] CEAB Accredits First Software Engineering Programs, June 2001. http://www.ccpe.ca/ccpe.cfm?page=softwareEngineering

[13] CEAB Accredited Software Engineering Programs, July 2002. http://www.ccpe.ca/ccpe.cfm?page=ceab_1998_10#S

[14] Computer Science Accreditation Council (CSAC) http://www.cips.ca/standards/accreditation/csac/

[15] Bachelor of Software Engineering (BSENG): Calendar Entry, 26 pages, Faculty of Engineering, University of Victoria, November 2002. http://www.cs.uvic.ca/~hausi/sedp/BSENG-Calendar-3.5.pdf

Appendix A: Letter of Intent

Appendix B: Comments from other institutions on the LOI

Appendix C: Letters in Support of BSENG Program

Appendix D: Calendar Course Descriptions

LETTER OF INTENT
TO OFFER A

BACHELOR OF SOFTWARE ENGINEERING

January 16, 2002

Faculty of Engineering
University of Victoria
Victoria BC

1. Institutional and Program Identification

The degree is to be offered by the University of Victoria. No formal arrangements with other institutions are required for this program. However, we plan to work towards a 'bridge' program that will allow students with an appropriate college diploma to enter third year. This will be modelled upon our highly successful bridge programs to Computer, Electrical and Mechanical Engineering. Transfer credit for college courses will also be available.

Program title: Software Engineering

Credential: Bachelor of Software Engineering

This degree is to be a joint offering by the Department of Computer Science and the Department of Electrical and Computer Engineering both in the Faculty of Engineering. Those Departments currently offer B.Sc. and B.Eng. degrees. The new degree name was chosen to distinguish this as an interdisciplinary degree program, and to reflect its uniqueness as being developed and administered jointly by the two departments. Note that this is the same name used for a similar program at the University of Waterloo. A faculty member will be appointed as Director of the Software Engineering Program.

This is to be an accredited degree with both the Canadian Engineering Accreditation Board (CEAB) and the Computer Science Accreditation Council (CSAC). The Association of Professional Engineers and Geoscientists of British Columbia (APEGBC) recognizes Software Engineering as a discipline and is registering Software Engineers as Professional Engineers.

Institutional contact:

Dean Michael Miller
Faculty of Engineering
University of Victoria
P.O. Box 3055
Victoria, BC V8W 3P6
(250) 721-8611 fax: (250) 721-8676
dean@engr.uvic.ca

2. Program Description

Software Engineering was identified as a key area in the January 2000 Faculty of Engineering Academic Renewal and Development Plan. Initiating a full undergraduate Software Engineering degree program was identified as a priority. The University has shown ongoing commitment to IT education, most recently as a partner in the *Doubling the Opportunity* proposal.[2]

[2] *Doubling the Opportunity: A Proposal to Double the Number of ICT Graduates at British Columbia Universities*, submitted to the Ministry of Advanced Education, Training and Technology by the University President's Council of British Columbia.

2.1 Brief Description

For more than 30 years, the term "software engineer" has been used so broadly that virtually anyone who has written a line of computer software can call herself/himself a "software engineer." It has been common practice for software vendors to disclaim any responsibility for the consequences of defects in the software that they market. Those who write such software have rarely been held accountable. The result has been the proliferation of software products that fail frequently, incurring significant costs to their users. For software used in safety-critical applications, such as medical, avionics, and automobile control systems, the risks are especially high.

In traditional Engineering disciplines, such as Electrical and Mechanical Engineering, the Canadian Engineering profession has long used (a) the accreditation process to ensure that engineers receive an acceptable level of education, and (b) professional certification to ensure that individual engineers can be held accountable for the products they design and build. The introduction of an accredited Software Engineering program is intended to bring a similar level of discipline and professional accountability to software developers. The industry has long recognized the need for such a program. Further, such programs have already been introduced at universities such as Concordia, McMaster, Ottawa, Western Ontario, and Waterloo.

The proposed Software Engineering program will be similar to other Engineering programs in its reliance on science and Mathematics as well as on practical design. However, the program will also cover the documentation, design, development, and verification of computer software. The focus will be both the application of Engineering methods to software, and the development of Engineering software applications. There will be special emphasis on three themes: (1) the analysis and testing of software for correctness, performance and usability, (2) the design of software to reduce the costs of maintenance (changes made after first installation), and (3) the design of software as part of a larger system.

This degree will consist of eight academic terms plus compulsory work experience.

2.2 Student Audience

This program is intended for students who seek professional careers in BC and Canadian companies involved in high technology, information technology and the development and application of software.

We anticipate an enrolment of 75 students per year (four years).

Currently, there is no full degree program designated as Software Engineering at BC post-secondary institutions. We presently offer an option in Software Engineering available to Computer Science students and a specialization in Software Engineering open to Computer Engineering students. At UBC, the Department of Computer Science offers a Software Engineering Option as does the Department of Electrical and Computer Engineering in the Computer Engineering degree. SFU offers a specialist program in

---

Detailed implementation and academic plans for each institution involved in the proposal are currently under development.

Software Engineering within the School of Computing Science. Many other institutions offer courses in Software Engineering.

2.3 Evidence of Market

The BC high-technology sector has seen remarkable growth in recent years, on Vancouver Island, in the Lower Mainland and elsewhere in the province. At ten percent growth per year, high technology is the fastest growing sector in British Columbia and now represents over 3% of our economy. The Vancouver Island Advanced Technology Centre (VIATEC) company directory lists 108 information technology companies on the Island, the vast majority, if not all, of which are actively involved in software research, development and application.

On a per capita basis, BC produces around 47% fewer Engineering and 25% fewer Computer Science Bachelor degree graduates than the respective Canadian averages. There is a clear need to increase the number of graduates in these areas in order to support economic growth in the province and to provide career opportunities for British Columbia's youth.

Software Engineering has long been established as a research discipline and Software Engineering has been taught within computer-related degree programs for many years. The first such course at UVic was offered in the late 70's. Software Engineering is now recognized as a discipline of significant importance in its own right and undergraduate degree programs have emerged elsewhere in Canada. There is to date no full undergraduate degree program in Software Engineering in B.C. yet the need for graduates educated in Software Engineering is great.

Software Engineering is now recognized as a professional discipline. The Association for Computing Machinery and the IEEE Computer Society established a joint steering committee for the establishment of Software Engineering as a profession in 1994. This committee has organized a number of task forces, including one addressing the education of Software Engineers. Closer to home, the Association of Professional Engineers and Geoscientists of BC recently began to register software engineers.

As evidence of very strong student interest in Software Engineering education, undergraduate Equivalent Enrolments Taught (EETs) in SENG courses have risen from 51 in 1998/99 (the year the SENG option and specialization were introduced) to 120 in 2001/02, an increase of 135%. For comparison over the same period, undergraduate EET enrolment across the Faculty rose 15.6%, while the undergraduate EET count at UVic rose 6.1% [3].

3. Admission and Transfer

This is to be a direct entry undergraduate program. The first year of the program is likely to be close to the first year in Computer Science and the Engineering programs. This means a student will be able to do first year and then choose between Computer Science,

[3] 2001/02 enrolment is the November 1, 2001 estimate published by UVic Institutional Analysis.

Computer Engineering or Software Engineering. Because of this, the well-established college transfer process will work for the new program. In addition a bridge program will be developed for Software Engineering.

Work experience challenge based on appropriate prior employment will be permitted. Course challenge based on prior learning will not be permitted but transfer credit for equivalent courses from other post-secondary institutions will be. This is consistent with current practice in the Faculty.

4. Program Resources and Timeline

4.1 Resources

The program will build on existing expertise in Software Engineering in the Faculty of Engineering and in particular on the experience gained in having offered program options in Software Engineering since 1998.

A significant number of new courses plus additional offerings of existing courses will be needed for this professional degree program. Additional faculty, support staff, teaching assistant, equipment and space resources will be required. These additions are part of the UVic Implementation Plan for the Doubling the Opportunity Proposal.

A total of 300 students (75 for each of four years) is equivalent to 300 EETs. It is anticipated that at least 70% of the program will be offered within Engineering resulting in 210 EETs in that Faculty. The remainder of the program, including general Engineering (ENGR) courses will be taught by other Faculties.

In general terms, the steady-state budget for the program within the Faculty of Engineering (in 2001 dollars) is

Salaries (Faculty & Staff including co-op)	$2,725,000
Teaching Assistants & Operating	$190,000
Equipment and software	$100,000
Total	**$3,015,000**

The program in steady-state will generate about 90 EETs outside the Faculty at an annual cost on the order of $500,000 (in 2001 dollars). Hence the total steady-state annual budget will be around $3,515,000 (in 2001 dollars).

A one-time amount of at least $200,000 will be required for equipment and software for the initial start-up phase of the program.

Implementation of the proposed degree program will require on the order of 2,610 m^2 in new space within the Faculty of Engineering for undergraduate laboratories and faculty, staff and technical support offices. That total space requirement has been included in the UVic Implementation Plan for the Doubling the Opportunity Proposal.

Note that the resources noted above are directly related to the implementation of the Bachelor of Software Engineering degree. Related costs and space requirements

regarding graduate students and enhanced research resulting from the new faculty hired for this program are accounted for in the Doubling the Opportunity Implementation Plan. We anticipate an increase of on the order of 160 weighted graduate student EETs as a result of the new faculty hired to offer the proposed degree program.

4.2 Effect on Other Programs

The students in this program will be over and above current program enrolments in keeping with the objectives of the Doubling the Opportunity proposal. The current Software Engineering option in Computer Science and Software Engineering specialization in Computer Engineering will be maintained as they use courses required for the full degree program. This will give students a choice to the extent to which they wish to emphasize Software Engineering in their studies.

4.3 Timeline

The intent is to start the program in September 2003. We anticipate a number of students will be able to transfer into the second and perhaps third year of the program when it starts because of the planned commonality of the junior years of the program with existing programs. We anticipate the full four-year program will be offered by 2005, and perhaps by 2004 depending on the final logistics of transfer arrangements and progress on faculty recruitment.

Consideration of the accreditation of the program by CEAB and CSAC will coincide with the first graduating class from the program.

Appendix B

Comments from other Institutions on the LOI

- Letter from Mr. Bill Parker, A/Director, Ministry of Advanced Education to Dr. David Turpin, President and Vice-Chancellor, University of Victoria
- Letter from Dr. Judith McGillivray, Vice-President, Learning and Provost, Kwantlen University College
- Letter from Dr. William R. Krane, Associate-Vice President, Academic, Simon Fraser University
- Letter from Dr. Rachael Donovan, Vice President, Academic, College of New Caledonia
- Letter from Dr. Martin Petter, Vice President, Education, North Island College

BRITISH COLUMBIA

April 23, 2002

File: 60555-20/IDPRC/LOI

Dr. David Turpin, President and Vice-Chancellor
University of Victoria
PO Box 1700 STN CSC
Victoria BC V8W 2Y2

Dear Dr. Turpin:

The Ministry of Advanced Education (AVED) has now completed its review of the University of Victoria's (UVIC) Letter of Intent for a Bachelor of Software Engineering, and I am pleased to advise you that UVIC is invited to proceed to the Full Program Proposal (FPP) stage for this program.

In developing the FPP the Ministry requests that UVIC provide further information on the consultation undertaken to date regarding accrediting bodies. In addition, UVIC is requested to clarify whether Prior Learning Assessment options will be available for students who have completed either formal or informal training in this subject area.

Once the FPP has been finalized, please forward 16 copies to AVED so that it may be scheduled for review at a subsequent meeting of the Degree Program Review Committee.

I look forward to receiving the FPP.

Sincerely,

Bill Parker
A/Director

pc: Jamie Cassels, Vice-President Academic and Provost
University of Victoria

Degree Program Review Committee Secretariat

RECEIVED
APR 26 2002
VICE-PRESIDENT
ACADEMIC AND PROVOST

Ministry of
Advanced Education

Public Institutions Branch
Post Secondary Education Division
2nd Floor, 835 Humboldt Street
PO Box 9877, Stn Prov Govt
Victoria BC V8W 9T6

Telephone: (250) 387-6189
Facsimile: (250) 952-6110

Kwantlen
UNIVERSITY COLLEGE

MAILING ADDRESS
12666 - 72nd Avenue
Surrey, BC
Canada V3W 2M8

TELEPHONE
(604) 599-2100
WEB SITE
www.kwantlen.ca

March 11, 2002

Office of the Provost and Vice President, Learning

Vicki Hocking
DPRC Secretariat
University Colleges Program Planning Branch
Ministry of Advanced Education
PO Box 9177, Stn. Prov. Gov't.
Victoria, BC

Dear Ms. Hocking:

Re: UVic Letter of Intent – Bachelor of Software Engineering

Dr. Abhijit Sen, Instructor in Computer Information Systems at Kwantlen University College, reviewed the above Letter of Intent. His comments are as follows:

"The University of Victoria's proposal to offer a Bachelor of Software Engineering program will fill a key gap in the offerings of Computing and Engineering programs in the province of British Columbia. Similar programs are already introduced in Ontario and Quebec, and it will provide students in our province with opportunities to be trained in important principles in the design, construction, and development of reliable software. Please note the following:

1. Page 2: Brief Description: 3rd paragraph: It says, *"focus will be both the application of engineering methods to software, and the development of engineering software applications"*. Should the focus be limited only to engineering software applications? The software engineering principles are equally applicable to the development of other systems such as financial and information systems. The new program must cater to the interests of diverse industries in British Columbia including hi-tech industries.

2. Page 2: Brief Description: 3rd paragraph: One of the areas in industries where it is difficult to find skilled people is in the area of Software Quality Assurance (which is more than software testing). Very few Software Engineering programs offer comprehensive training in this area. Any new Software Engineering program must provide students with in-depth training in the engineering principles of Software Quality Assurance, and expose the students with a variety of tools to implement software quality. Industry has much to gain if the intended program has this as one of the major themes of emphasis. This will make the program much more unique.

3. Page 3-4: Admission and Transfer: In this section it is stated *"Course challenges based on prior learning will not be permitted"*. Software Engineering by its nature is an applied discipline, and prior learning assessment (PLA) is a well established mechanism to grant formal recognition for knowledge and skills already acquired by prospective students. Perhaps prior learning assessment could be included as an option so that access to the program would be available to many stakeholders."

In summary, the University of Victoria's proposal is an excellent one and we are hopeful that it will receive approval.

Thank you for the opportunity to respond to this degree proposal.

Yours truly,

Judith McGillivray

Judith McGillivray
Vice President, Learning, and Provost

copy

√ Dr. Michael Miller, Dean
Faculty of Engineering
University of Victoria

SIMON FRASER UNIVERSITY

ASSOCIATE VICE-PRESIDENT, ACADEMIC

8888 UNIVERSITY DRIVE
BURNABY, BRITISH COLUMBIA
CANADA V5A 1S6
Telephone: (604) 291-4686

March 13, 2002

Ms. Vicki Hocking
DPRC Secretariat
Post-Secondary Education Division
Ministry of Advanced Education, Training and Technology
1st Floor 835 Humboldt Street
Victoria, BC V8V 9H8

Dear Ms. Hocking:

Subject: Bachelor of Software Engineering, University of Victoria

We have reviewed the Bachelor of Software Engineering proposal from the University of Victoria, and attach comments from Dr. Robert Cameron, Associate Dean, Faculty of Applied Science and Dr. Ze-Nian Li, Director, School of Computing Science.

When the final program proposal document becomes available, we would appreciate receiving a copy of it.

Thank you for the opportunity to comment on this proposal.

Yours sincerely,

William R. Krane, PhD
Associate Vice-President, Academic
Simon Fraser University

c: Dr. Valerie Kuehne, Associate Vice-President Academic, UVic

encl.

more expensive and/or produce fewer graduates per year. Certainly, the proposed budget for this program seems well above that required for software engineering options within computing science programs.

In order to address these issues, it may be that the traditional degree program approval process ought to be augmented by appropriate consultation with the information technology sector on the general need for certification of computing professionals and the accreditation of programs producing them.

Comments received from:
Dr. Ze-Nian Li, Director, School of Computing Science

My understanding is that both CSAC and CEAB are accrediting some software programs (Ottawa, Western, USask, etc.). Most people seem to accept the fact that both CS and EE (ECE) will be offering these programs whether independently or jointly, and more and more of them will be accredited by CSAC and/or CEAB. On that, the UVic proposal seems just fine.

The most contentious issue is the "exclusive right of practising" which most Engineering Acts currently stipulate. Apparently ACM and IEEE are looking into this, and nationally there were lots of activities and events (including the Newfoundland lawsuit). On that, I think Rob's message is a good cautionary note.

College of New Caledonia

March 13, 2002

BY FAX: (250) 356-8851

Vicki Hocking, DPRC Secretariat
Ministry of Advanced Education, Training & Technology

Response to New Program Proposal For:
Bachelor of Software Engineering
University of Victoria

The Dean and the faculty in the Science & Technology Department at the College of New Caledonia have reviewed the proposal and strongly support the establishment of this new program.

Sincerely,

R. Donovan

Rachael Donovan
Vice President Academic

cc: > Dr. Valerie Kuehne — BY FAX: (250) 721-7216
Associate V-P Academic, Uvic

> Michael Miller — BY FAX: (250) 721-8676
Dean of Faculty of Engineering, UVic

> Ben Malcolm
Dean of Science & Technology, CNC

Prince George Campus
3330 - 22nd Avenue
Prince George, B.C.
V2N 1P8
Tel: (250) 562-2131
Fax: (250) 561-5816
Board & Executive Fax:
(250) 561-5829

Lakes District Campus
Box 5000
Burns Lake, B.C.
V0J 1E0
Tel: (250) 692-1700
Fax: (250) 692-1750

Mackenzie Campus
Box 2110
Mackenzie, B.C.
V0J 2C0
Tel: (250) 997-4333
Fax: (250) 997-3779

Quesnel Campus
488 McLean Street
Quesnel, B.C
V2J 2P2
Tel: (250) 991-7500
Fax: (250) 991-7502

Nechako Campus
R.R. #2
Vanderhoof, B.C
V0J 3A0
Tel: (250) 567-3200
Fax: (250) 567-9584

UNIVERSITY OF VICTORIA
VICE-PRESIDENT ACADEMIC AND PI

Office of the Vice President, Education
Komoux Hall
2300 Ryan Road
Courtenay, BC V9N 8N6
tel 250.334-5223 | fax 250.334-5274
email mpetter@nic.bc.ca

TO: Dean Miller

DATE: Mar. 15/02

- ☑ For your information
- ☐ For you to handle, please
- ☐ For your comments, please
- ☐ Let's discuss this, please
- ☐ As request
- ☐ Return, ple
- ☐ Returned v
- ☐ Keep or de

15 March, 2002

Ministry of Advanced Education, Training & Technology
PO Box 9177 Stn Prov Govt
Victoria, BC
V9W 9H8

RECEIVED
288
MAR 18 2002
OFFICE OF THE DEAN
FACULTY OF ENGINEERING

Attention: Vicki Hocking
DPRC Secretariat

Dear Ms. Hocking:

Re: **Letter of Intent**
Bachelor of Software Engineering – University of Victoria

We are in receipt of the Letter of Intent for the above referenced New Degree Program proposal forwarded to our office on February 15, 2002. The Letter of Intent was reviewed by Brigid Walters, Dean of Academic Programs. The following comment was made:

- We would like clarification on transfer options for College students. Otherwise we find this excellent.

Thank you for providing us with the opportunity to respond with any concerns or comments we may have on the above Letter of Intent.

Yours truly,

D Goudy for Martin Petter
Vice President, Education

:dg

cc: B. Walters, Dean, Academic Programs, NIC
Dr. Valerie Kuehne, Associate Vice-President Academic, UVic

Copy to Dean Miller 15 March 02

V:\VP\LOI\Bachelor of Software Engineering UVIC.doc

Mission

rth Island College's rpose is to ensure elevant learning rough accessible, uality education.

orth Island College is dedicated to supporting economic, social, ultural and nmental su...inability of e College region. is committed to enhancing the quality of life, zenship, community participation and employment portunities of the idents of its region.

orth Island College responds to the irners in its region t but may operate ovincially, nationally nd internationally.

Values

Our mission rests on these values:
Access
Relevance
Quality
Student Success
ucational Leadership
Responsiveness
earning
untability
Ethical Practice
sitive Environment
logical Sustainability
ultural Sustainability

Appendix C

Letters in Support of University of Victoria BSENG Program

- David Clark, VP Engineering and Former Board Chair of WestMOST MacDonald, Dettwiler and Associates (MDA) [http://www.mda.ca, dpc@mda.ca]
- William C. Cooke, Chief Executive Officer, Vancouver Island Advanced Technology Centre (VIATEC), Victoria [http://www.viatec.ca/, bcooke@viatec.ca]
- Victor Jones, President and Executive Director, BC Advanced Systems Institute (ASI), Vancouver [http://www.asi.bc.ca]
- Anatol Kark, National Research Council (NRC), Ottawa [http://www.nrc.ca, anatol.kark@nrc.ca]
- Richard Peacocke, Nortel Networks, Ottawa [http://www.nortelnetworks.com/, richard@nortelnetworks.com]
- Mr. Joe Wigglesworth, Program Director, IBM Centre for Advanced Studies (CAS), Toronto [http://www.cas.ibm.ca/toronto/people/members/joe.shtml, wiggles@ca.ibm.com]
- Dr. Dennis Smith, Carnegie Mellon Software Engineering Institute (SEI), Pittsburgh, USA [http://www.sei.cmu.edu, dbs@sei.cmu.edu]
- Wolfgang Strigel, Director Software Productivity Center Inc. (SPC), Vancouver [http://www.spc.ca, strigel@spc.ca]

MACDONALD DETTWILER®

September 5, 2002

Professor J.W. Provan
Acting Dean
Faculty of Engineering, University of Victoria
PO Box 3055
Victoria, BC
V8W 3P6

To Whom It May Concern:

This letter indicates my strong support for the Faculty of Engineering at the University of Victoria to offer a Bachelor of Software Engineering Program. In my opinion, degrees in the engineering of software are now as, if not more, important to the growth of the economy of British Columbia, and to the well being of its citizens, than those in the allied fields of computer science and electrical engineering. Software is now all pervasive in our society, and we can no longer afford the luxury of it being developed in an ad-hoc manner with little or no attention paid to its quality. This is especially true in areas where human life is at stake – air traffic control, medicine, avionics – and where software is embedded as the control function in larger systems valued in the tens of millions of dollars – satellites, process control, robotics, military systems. Ultimately it is my belief that software engineers will have to be certified by a recognized body if they are to be allowed to develop such applications, and a degree in software engineering will be necessary (but not sufficient) to achieve that.

I have looked at the proposed curriculum for this degree from the Faculty of Engineering and find it well suited to the training and development of software engineers for industry and academia. I hope you are able to support it fully and hence enable the launch of the program by this time next year.

Yours faithfully,

David F. Clark

David Clark
Director of Engineering
MacDonald Dettwiler and Associates
Richmond
British Columbia

DC:sd

Vancouver Island Advanced Technology Centre

VIATeC

850 Courtney Street
Victoria, BC V8W 1C4
Ph: 250.953.6680 Fax: 250.953.6679
email: mail@viatec.ca website: http://www.viatec.ca

26 September, 2002

Dr. J.W. Provan, Professor and Acting-dean
Faculty of Engineering, Office of the Dean
University of Victoria
PO Box 3055
Victoria, BC V8W 3P6

Dear Dr. Provan:

Re: New Bachelor of Software Engineering Program, Faculty of Engineering University of Victoria

Thank you very much for forwarding your proposal for the creation of a new undergraduate degree program in Software Engineering at the University of Victoria for our review and comments. We have had an opportunity to review it and would like to add our support to this very worthwhile initiative.

As you are aware, VIATeC is a non-profit advanced technology industry association with nearly 600 members on Vancouver Island (a region generating nearly $1Billion annual revenues through the technology sector). A large number of our technology producing member companies would benefit from a program such as this that produced high quality graduates in our local community. It would be of value not only to our IT companies but also to our hardware product companies as a significant portion of product functionality is dependant on embedded software.

The University of Victoria has an excellent reputation for delivering high quality degree programs combined with a Co-op format. A program such as is proposed would be a valuable complement to your existing engineering programs and would be welcomed in our technology community.

I trust you will be successful in having the program approved and we would be pleased to contribute in any way possible to move this initiative forward. Please contact me at your convenience if you need any further support or assistance.

Sincerely

W. C. Cooke, P.Eng.
CEO

A S I

BC Advanced Systems Institute

#450 – 1122 Mainland Street
Vancouver, British Columbia, Canada V6B 5L

Tel: [604] 689-0551
Toll Free: 1-800-501-3388
Fax: [604] 689-4198
E-mail: asi@asi.bc.ca
Web: www.asi.bc.ca

October 29, 2002

1070

Dr. James W. Provan
Professor and Acting Dean
Faculty of Engineering
University of Victoria
PO Box 3055
Victoria, BC
V8W 3P6

Dear Dr. Provan,

Re: New Bachelor of Software Engineering Program, Faculty of Engineering

From our perspective, your proposal to initiate a Bachelor of Software Engineering degree should be endorsed and we add our support . Improved computer training and a professional degree for students in BC fits well with the objectives of ASI, which include the promotion of collaborative efforts between research and industry and the commercialization of new technologies.

Establishing a degree program to give focus to software as an engineering discipline would likely be attractive to students. Your proposed program would also give further deserved recognition to the integral aspect which software plays in products across industry lines and the need to establish recognized standards and accreditation.

On this topic, I spoke briefly with John Watson, outgoing president of APEG, BC. APEG has given some thought to this matter and he has offered to arrange a meeting with you at the upcoming APEG Annual Meeting in Victoria later this month.

We look forward to hearing of your progress.

Yours truly,

Victor J E Jones
President

National Research Council Canada
Institute for Information Technology

Conseil national de recherches Canada
Institut de technologie de l'information

NRC·CNRC

September 23, 2002

Prof. J.W. Provan
Acting Dean, Faculty of Engineering
University of Victoria
PO Box 3055
Victoria, BC Canada
V8W 3P6

Dear Professor Provan,

Re: Letter in support of the Bachelor of Software Engineering Program, (BSENG) Faculty of Engineering, University of Victoria.

It is my great privilege and pleasure to provide you with a letter of support for the proposed Bachelor of Software Engineering Program at your Faculty.

I consider your proposal vital not only to the well being of the University of Victoria and British Columbia, but also to the economy of the entire country. In a very short period of time, software has become an integral and vital part of everyday life. However, the tools that allow for the systematic, disciplined and consistent development and maintenance of software have not kept pace with the demands placed on software or the requirements for software quality. The demand for properly trained software professionals has long outpaced the ability of the educational system to provide those professionals. The proposed program at the University of Victoria, other programs that served as a model for it and the programs of the future that will model themselves on your program will go long way towards solving that imbalance.

In my opinion, Canada has relied too much on favourable exchange rates for the Canadian dollar as an advantage in the global economy. That "advantage" is predicted to disappear. We must strive to base our advantage on the enhanced quality of Canadian software professionals, to avoid the "neither cheap nor good" possibility. Again, your Program will be a part of the solution.

.../2

Ottawa, Canada
K1A 0R6
Fax: 613 952-7151
Cable Address
Research Ottawa

Ottawa, Canada
K1A 0R6
Télécopieur: 613 952-7151
Adresse télégraphique
Research Ottawa

Canada

2/ Prof. J.W. Provan

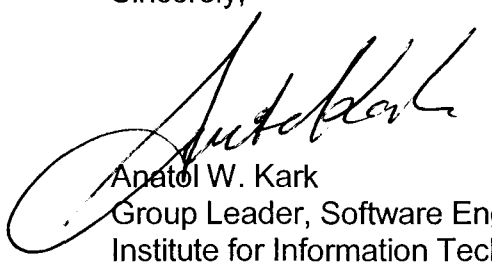
Any engineering discipline requires a support of basic sciences and empirical experiences. The proposed Program at your Faculty is well positioned in that respect. The strong Computer Science Department at the University of Victoria, and the excellent understanding of Software Engineering issues evidenced by the proponents of the proposal – as shown by the proposal itself - bodes well for the future of the Program. I am especially pleased that the proposed program is meant to provide for a mandatory 16 months co-op experience for your students. Practical experience is essential in the training process of any Engineering discipline.

I found that the proposal addressed most of the subjects, which I would consider vital in the training of Software Engineers. To a large extent, it follows recommendations contained in the Software Engineering Body of Knowledge (SWEBOK) and in an upcoming Software Engineering Education Knowledge (SEEK). Both of these documents are sponsored and endorsed by IEEE.

One of the goals of the Software Engineering Group at NRC is to contribute to raising software development tools and methodologies to engineering levels. With that mutual goal in mind, we have collaborated with like-minded faculty members on your faculty. We are looking forward with great anticipation to broadening that collaboration once your Program is established.

Please do not hesitate to contact me at any time if you require any further assistance. Wishing you success in your endeavour,

Sincerely,



Anatol W. Kark
Group Leader, Software Engineering Group
Institute for Information Technology
National Research Council Canada

Cc: Dr. Andrew Woodsworth, Director General, IIT, NRC

Nortel Networks Limited
PO Box 3511, Station C
Ottawa ON K1Y 4H7 · Canada
MailStop 04554E05
Tel(613) 765-2629
Fax (613) 765-1737

NORTEL NETWORK

30 September, 2002

James W. Provan
Professor and Acting Dean
Faculty of Engineering
University of Victoria
PO Box 3055
Victoria, B.C.
V8W 3P6

New Bachelor of Software Engineering Program, Faculty of Engineering

Dear Prof. Provan,

I would like to voice my strong support for the proposed new undergraduate degree program in Software Engineering at the University of Victoria.

Need for program

Canada's software sector employment doubled from 1995 to 2000. There are over 200,000 IT staff currently employed, and despite the economic downturn there is still a shortage in some areas, particularly of experienced staff in design and project management. A knowledge and innovation-based economy—Canada's target—will never be achieved without a steady supply of highly skilled staff to support economic growth. The proposed program will help achieve that.

Strength of proposed program

The proposal builds on a solid established base. University of Victoria already has a distinguished group of researchers and teachers in Software Engineering. I am familiar with much of their work, partly from various review activities that I have undertaken for NSERC. The Software Engineering option at University of Victoria has been successful, and this proposal takes that option a logical step forward.

Fusion of computer science and engineering principles is vital for contributing to industrial software development. The proposed program is well-designed to attain the blend of science, engineering, and external work experience required; and I am happy to see it working from the IEEE/ACM curriculum recommendation, and choosing to meet all the requirements of CEAB and CSAC.

Personal perspective

I have been working in software development at Nortel Networks since 1977, with over half my career there engaged in software engineering and quality assurance activities on large systems—switching, and fibre-optic transmission and switching. The software developed for them is in the tens of millions of lines of code. The other part of my career at Nortel has been in software-intensive technology areas such as artificial intelligence applications in telecommunications, office communications, and speaker recognition. Prior to my graduate work in Canada (Master's and PhD) I was with IBM UK for several years mainly in engineering and real-time scientific applications.

If I can be of any more assistance in advancing the proposed program please do not hesitate to contact me.

Sincerely,

Richard Peacocke

Dr. Dick Peacocke

Optical Networks, Development Process Solutions

richard@nortelnetworks.com

Voice: (905) 413-3856
Fax: (905) 413-4927
Email: wiggles@ca.ibm.com

IBM Canada Ltd.
8200 Warden Ave.
Markham, ON L6G 1C7

Dean James W. Provan
Faculty of Engineering
University of Victoria
P.O. Box 3055, Victoria, BC
V8W 3P6

October 11, 2002

Dear Dean Provan,

Thank you very much for the opportunity to review your document *Proposal to Establish a Bachelor of Software Engineering Degree*. I've read it over and I must say that I'm very impressed with it. It is clear that your proposal is built upon a well-researched curriculum that will be delivered by highly-qualified faculty members. Because of the importance of software engineering to the economic development of Canada, I wholeheartedly support the establishment of this academic program because it is clear that it will advance the state of the art and practice of software engineering in Canada.

The team here at the Centre for Advanced Studies (CAS) at the IBM Toronto Laboratory has a long history of collaboration with the software engineering researchers at the University of Victoria, notably Dr. Hausi Müller and Dr. Peggy Storey who are both active CAS Visiting Scientists. Their guidance has been very helpful to development projects here in the Toronto Laboratory through the years, and I know that same experience will ensure the success of your Bachelor of Software Engineering proposal. I also know from attending conferences such as the International Conference on Software Engineering that the software engineering researchers at the University of Victoria are regarded as world leaders in their field.

I wish you tremendous success with this proposal.

Yours truly,

J. Wigglesworth

Mr. Joe Wigglesworth, P.Eng
IBM Toronto Laboratory
Manager, Centre for Advanced Studies

Carnegie Mellon
Software Engineering Institute
Pittsburgh, Pennsylvania 15213-3890

September 30, 2002

Professor J. W. Provan, Professor and Acting Dean
Faculty of Engineering, Office of the Dean
University of Victoria
PO Box 3055
Victoria, BC, Canada V8W 3P6

Dear Professor J. W. Provan:

I strongly support the proposal for the Bachelor of Software Engineering Program at the University of Victoria. Over the past several decades, software has become a dominant driver for economic growth. The need for strong programs in software engineering is critical to the future of the Canadian economy, as well as the economies of other developed societies.

However, there remains a shortage of highly trained individuals. In my work I have unfortunately come across far too many software programs that have been ill-conceived, leading to the waste of millions of dollars. Such failures would have had a much greater probability of success if decision-makers had been grounded in strong software engineering theory and practice.

I believe that the proposed Bachelor of Science in Software Engineering Program will help toward filling that gap. The proposed program is well thought out and rigorous in its approach. The current faculty has established world class software engineering research projects, and they have an international reputation for high quality. These faculty members have pioneered breakthroughs in software engineering methods and tools, and they offer a strong foundation for the degree program.

In summary, this program fills an important need, is well thought out, and has an excellent faculty in place as a foundation for carrying it out.

Sincerely,

Dennis Smith

Phone: 412 / 268-7700 • FAX: 412 / 268-5758 • Web: http://www.sei.cmu.edu

Software Productivity Center Inc.

2002-09-27

James W. Provan
Acting Dean of Engineering
University of Victoria

Subject: Bachelor of Software Engineering Degree

Dear Dr. Provan,

With this letter I would like to express my strong support for the proposed Bachelor program in Software Engineering. I have reviewed the proposal (Version 4.12, September 21, 2002) and find the curriculum very attractive and pertinent for the needs of the industry. This opinion is based on SPC's experience in consulting with software development organizations across North America. The proposed program is designed to build essential skills and knowledge, which should serve the student as a general foundation for further specialized studies, or to prepare them to enter the work force with confidence.

The mix of foundation course and specialized topics appears well balanced. Without knowledge of the detailed course contents it may be preposterous to suggest some potential additions or alternatives. However, I feel that it would be useful to ensure that the following areas are well covered:

- Software development process across the full lifecycle
- Testing (I call this topic the "stepchild" of software engineering since, sadly, most curricula ignore this rather important discipline)
- Team psychology: arguably, software development is a 'team sport', and professionals in our industry are asked to be team players, whether they are part of large teams or involved in pair programming. The nature of technically oriented people is often that of individualists and the success of projects and teams relies on effective collaboration.

I realize that these topics may be covered in some electives but I believe these skills should be part of the foundation courses.

I also applaud you for the requirement to gain practical industry experience through your COOP program. Its flexibility gives the students the option to gain in depth experience or to sample different environments. It is also a valuable source of capable workers for the industry and an excellent trial ground to find future recruits.

In summary, I believe this program will be extremely valuable to produce the knowledge and skill set we need in our industry. It should also be a very popular program that will attract skilled students and outstanding candidates as faculty members.

Sincerely,

Wolfgang Strigel
President

T 604.662.8181 **F** 604.689.0141 **E** info@spc.ca
Suite 560, 1122 Mainland Street, Vancouver, BC, Canada V6B 5L1

Appendix D

Proposed University of Victoria Bachelor of Software Engineering (BSENG) Degree Program

Calendar Course Descriptions V6.0—November 8, 2002

Introduction

This document contains the calendar course descriptions of the proposed University of Victoria *Bachelor of Software Engineering (BSENG)*. To get an overview of the courses, please consult the *BSENG Formal Degree Program Proposal*, the *BSENG overview matrix*, the *BSENG prerequisite structure,* and the *BSENG Calendar Entry*. To gain insight on how the *BSENG curriculum* was developed, please consult the *BSENG Curriculum Executive Summary*.

These course descriptions are a combination of University of Victoria calendar descriptions (i.e., SENG, CSC, CENG, and ELEC), IEEE/ACM Computing Curricula 2001 descriptions (i.e., Steelman Draft, August 1, 2001), revised course descriptions, and new course descriptions.

Table of Contents

Required courses

CAP: Capstone Project
LA: Linear Algebra (Matrix Algebra for Engineers)
WE: Web Engineering
PS: Introduction to Probability and Statistics
HCI: Human-Computer Interaction
OSDC: Operating Systems and Distributed Computing
DB: Databases
NET: Networks
RT: Real-time Systems
CALC 1: Calculus 1
CALC 3: Calculus 2
SYSDYN: System Dynamics
SAS: Signal and Systems
CTRL: Control Systems
SEC: Security Engineering
DS 1: Discrete Structures 1
DS 2: Discrete Structures 2
BUS 1: Engineering Economics and Entrepreneurship
BUS 2: Engineering Planning and Management
SOCIAL: Social and Professional Issues

Engineering Electives

ARCH: Software Architecture
CBSE: Component-Based Software Engineering
CSCW: Computer-Supported Collaborative Work
FTC: Fault Tolerant Computing
CG: Computer Graphics
MMS: Multimedia Systems
ALG 2: Algorithms and Data Structures 2
ALG 3: Algorithms and Data Structures 3
CC: Compiler Construction
CON: Concurrency
PL: Programming Languages
AI: Artificial Intelligence
PATREC: Pattern Recognition
ROBOT: Robotics
IKM: Information and Knowledge Management
NC: Network-Centric Computing
DC: Distributed Computing
DSP: Signal Processing
COM: Digital Communications
WMC: Wireless and Mobile Computing
NUM: Numerical Methods
NA1: Numerical Analysis I

NA2: Numerical Analysis II

ORLP: Operations Research: Linear Programming
ORSIM: Operations Research: Simulation
MIN: Data Mining
MIRCO: Microprocessor Systems
CSA: Computer Systems and Architecture
ERGO: Ergonomics
MCOM: Mobile Communications
AOOD : Advanced Object Oriented Design
SP : Software Process
MSD : Management of Software Development
TOP : Topics in Software Engineering

Calendar Course Descriptions

Required Courses

SE 1: Object-Oriented Programming
Introduces the fundamental concepts sequential and object-oriented programming. Through the study of object design, this course also introduces the basics of human-computer interfaces, and the social implications of computing, along with significant coverage of software engineering.
Prerequisites: none

SE 2: Object-Oriented Design and Methodology
Continues the introduction to object-oriented programming begun in SE 1, with an emphasis on algorithms, data structures, software engineering, and the social context of computing.
Prerequisites: SE1, LA

SE 3: Software Architecture and Development Methods
Provides an introduction to software architecture and development methods including aspects of object-oriented analysis, design and development to create medium-scale applications. Topics include software architecture and components, object-oriented design and metrics, component integration, interfaces, component libraries and reuse, exception handling, serialization, testing, and project management.
Prerequisites: SE2, WE

SE 4: Systems Programming and Middleware
Provides an introduction to systems and middleware programming using programming and scripting languages. Topics include interacting with the various components of an operating system, end-user programmable tools, and the Web. Control, data, and presentation integration mechanisms as well. Web interoperability through Web standards, protocols, and services.
Prerequisites: SE3, CAS

SE 5: Requirements Engineering and Formal Specification
Combines a range of topics integral to the design, implementation, and testing of a medium-scale software system with the practical experience of implementing such a project as a member of a programmer team. Introduces formal methods, requirements engineering, specifications, software life cycle models.
Prerequisites: ALG1, SE4, DS2 or ALG1, CENG 245, registration in CENG

SE 6: Software Evolution
Introduces problems and solutions of long-term software maintenance/evolution and large-scale, long-lived software systems. Topics include software engineering techniques for programming-in-the-large, programming-in-the-many, legacy software systems, software architecture, software evolution, software maintenance, reverse engineering, program understanding, software visualization, advanced issues in object-oriented programming, design patterns, antipatterns, and client-server computing. This course culminates in a team project.
Prerequisites: SE5

SE 7: Embedded Systems
Characteristics and design of embedded systems. Formal models and specification languages for capturing system behaviour. Techniques for specification, exploration and refinement. System partitioning and hardware/software co-design. Tools for validation, verification, and simulation. Quality and performance metrics. Embedded real-time systems are pervasive in today's world (e.g., telecommunications systems, consumer electronic products, automotive systems, and aerospace systems).
Prerequisites: SE6

SE 8: Software Quality Engineering
This course emphasizes software quality engineering as an integral facet of development, from requirements through delivery and maintenance. The students will learn how to choose appropriate quality goals and select, plan, and execute quality assurance activities throughout development and evolution to predictably meet quality and schedule goals. They will learn how quality assurance can be incorporated into process improvement feedback loops that amplify the ability of an organization to cost-effectively prevent and detect faults.
Prerequisites: SE6

CAP: Capstone Project
Offers students the opportunity to integrate their knowledge of the undergraduate software engineering curriculum in a significant design experience which is based on the knowledge and skills acquired in earlier course work.
Prerequisites: SE6

MECHSYS: Mechanical Systems for Engineers
Introduce engineering students to systems built by mechanical engineers and modes of thinking used by these engineers through the teaching of selected subjects with lectures by faculty and invited speakers from industry, films, and laboratory demonstrations. As an introductory course students will be exposed to fundamentals of engineering design, engineering ethics, problem solving methods demanding the application of fundamental engineering principles and checking of solutions. Introduce students to the terminology, mathematics, and methods used in mechanical engineering. The lectures also discuss career opportunities in mechanical engineering and related fields, emerging technologies, and the cross-disciplinary nature of engineering. The course features a significant project. Motivating engineering students is a key aspect of this course.
Prerequisites: none

ELECSYS: Electrical Systems for Engineers
Introduce engineering students to systems built by electrical and computer engineers and modes of thinking used by these engineers through the teaching of selected subjects with lectures by faculty and invited speakers from industry, films, and laboratory demonstrations. As an introductory course students will be exposed to fundamentals of engineering design, engineering ethics, problem solving methods demanding the application of fundamental engineering principles and checking of solutions. Introduce students to the terminology, mathematics, and methods used in electrical engineering. The lectures also discuss career opportunities in electrical engineering and related fields, emerging technologies, and the cross-disciplinary nature of engineering. The course features a significant project. Motivating engineering students is a key aspect of this course.
Prerequisites: PHYS

DS 1: Discrete Structures 1
Introduces the foundations of discrete mathematics as they apply to computer science, focusing on providing a solid theoretical foundation for further work. Topics include functions, relations, sets, simple proof techniques, Boolean algebra, propositional logic, digital logic, elementary number theory, and the fundamentals of counting.
Prerequisites: CALC1

DS 2: Discrete Structures 2
Continues the discussion of discrete mathematics introduced in DS 1. Topics in the second course include predicate logic, recurrence relations, graphs, trees, matrices, computational complexity, elementary computability, and discrete probability.
Prerequisites: DS1

CAS: Computer Architecture and Assembler Programming
Introduces students to the organization and architecture of computer systems, beginning with the standard von Neumann model and then moving forward to more recent archictural concepts.
Prerequisites: SE2

ALG 1: Algorithms and Data Structures
Introduces formal techniques to support the design and analysis of algorithms, focusing on both the underlying mathematical theory and practical considerations of efficiency. Topics include asymptotic complexity bounds, techniques of analysis, algorithmic strategies, and an introduction to automata theory and its application to language translation.
Prerequisites: SE2, DS1

DD: Digital Design
This course develops a structured design methodology for the design of complex digital systems. This is achieved by using a systems level approach to the development of a digital design. An introduction to the use of suitable CAD tools is given in the laboratories whilst covering the more theoretical aspects associated with logic design in the lectures.
Prerequisites: CAS

AFL: Automata Theory and Formal Languages
A survey of formal models and results that form the theoretical foundations of computer science; typical topics include regular and context-free languages, finite automata, Chomsky hierarchy, Turing machines, undecidable problems, and computational complexity.
Prerequisites: ALG1, DS2

LA: Linear Algebra
Complex numbers; matrices and basic matrix operations; vectors; linear equations; determinants; eigenvalues and eigenvectors; linear dependence and independence; orthogonality.
Prerequisites: none

WE: Web Engineering
Introduces students to the world of computing and communications through the World-Wide Web. SE 1 programming background is required, students will learn some programming through scripting languages. Topics include security, privacy, history, multimedia technologies, HCI, network management, electronic commerce.
Prerequisites: LA

PS: Introduction to Probability and Statistics
Descriptive statistics; elementary probability theory; random variables, discrete and continuous probability distributions, expectation, joint, marginal and conditional distributions; linear functions of random variables; random sampling and sampling distributions; point and interval estimation; classical hypothesis testing and significance testing. The mathematical foundations of statistical inference will be introduced and illustrated with examples from a variety of disciplines.
Prerequisites: none

HCI: Human-Computer Interaction
Presents a comprehensive introduction to the principles and techniques of human-computer interaction.
Prerequisites: PS, SE3

OSDC: Operating Systems and Distributed Computing
Introduces the fundamentals of operating systems together with the basics of networking and communications.
Prerequisites: ALG1, SE4 or ALG1 and third-year standing in CENG

DB: Databases
Introduces the concepts and techniques of database systems.An introduction to the use and operating principles of database management systems. Topics to be covered include: data entities and relationships; data modeling using Entity-Relation Diagrams: hierarchical, network and relational models of databases; query languages; physical representation of data in secondary storage; relational algebra and calculus as applied to the design of databases; security and integrity in the context of concurrent use; and basic ethical issues associated with database design and use.
Prerequisites: OSDC, SE4

NET: Networks
Introduces the structure, implementation, and theoretical underpinnings of computer networking and the applications that have been enabled by that technology.
Prerequisites: OSDC, SEC

RT: Real-time Systems
Fundamental issues in design of real-time operating systems and application software. Typical topics include: hard real-time scheduling, interrupt driven systems, process communication and synchronization, language requirements for real-time systems, decomposition of real-time requirements into process models, and case studies. A project involving design, implementation and testing of a real-time executive and real-time application software will also be included.
Prerequisites: SE7

CALC1: Calculus 1
Review of analytic geometry; functions and graphs; limits; derivatives; techniques and applications of differentiation; antiderivatives; the definite integral and area; logarithmic and exponential functions; trigonometric functions; Newton's, Simpson's and trapezoidal methods.
Prerequisites: none

CALC2: Calculus 2
Volumes; arc length and surface area; techniques of integration with applications; polar coordinates and area; l'Hospital's rule; Taylor's formula; improper integrals; series and tests for convergence; power series and Taylor series; complex numbers.
Prerequisites: CALC1

SYSDYN: System Dynamics
System definition and properties. Continuous and discrete systems. Linearity, causality, determinism, equivalence. Modeling of linear time-invariant systems using differential and difference equations with applications. Solutions of differential and difference equations. Laplace transforms and the representation of signals and systems.
Transfer functions and block diagrams. Fourier transforms and Fourier series in spectral analysis. Interrelation between the Fourier and Laplace transforms. Extensive use of MATLAB including simulation of engineering systems and applications.
Prerequisites: CALC2

SAS: Signal and Systems
Continuous time signals and waveform calculations. The Fourier series in the analysis of periodic signals. The impulse and other elementary functions. Resolution of signals into impulse and unit step functions. The Fourier transform in spectral analysis. Functions of a complex variable. Analytic functions. Partial fractions. The Laplace transform in the representation of signals. Interrelation between the Fourier and Laplace transforms. Design project using Matlab.
Prerequisites: SYSDYN, ELECSYS

CTRL: Control Systems
To introduce the basics of design and analysis of control systems. Principles of control: block diagrams, transfer functions, open and closed loop systems, linear-time invariant systems, Bode Plot and Nichols' Chart. Performance specification and estimation: stability criterion, Routh-Hurwitz and Nyquist stability criteria; root locus methods; steady state errors, transient performance. Simple design methods. Discrete systems: Z-transform, stability criterion, discrete continuous equivalence, sampling interval considerations. Automation: the design process, design specification, technological alternatives, economics, sensor systems, actuation systems, interfacing, signal conditioning, DC servos, Proportional-Integral-Derivative (PID) controllers; lead and lag compensators; robust design. Microprocessor based control systems. Product design. Aspects of robot system design.
Prerequisites: SYSDYN

SEC: Security Engineering
This course presents the fundamentals of contemporary computer security and cryptology. Topics included an overview of computer security, protection, disaster planning, and recovery. Risk analysis and security plans. Basics of cryptography. Public key cryptography and protocols. Security models, kernel design and systems testing. Database, network and Web security. The course discusses applications which need various combinations of confidentiality, availability, integrity and covertness properties; mechanisms to incorporate these properties in systems. The course also deals with policy and legal issues.
Prerequisites: OSDC, SE5

BUS1: Engineering Economics and Entrepreneurship
Macroeconomic principles: money, interest rates, growth. Microeconomic principles: demand and supply, production, consumer utility and elasticity. Net present value, equivalence, rate of return. Public vs private sector cost-benefit analysis, externalities, risk and uncertainty. Industry and innovation life cycles. Entrepreneurship: starting and running a business, identifying market need, researching financial viability, and resource requirements (financial, human, technical).
Prerequisites: none

BUS2: Engineering Planning and Management
An introduction to and overview of finance and accounting for engineering management. Topics include basic accounting concepts and terminology; preparation and interpretation of financial statements; and uses of accounting information for planning, budgeting, decision-making, control, and quality improvement. Price and output decisions. Choosing among alternative inputs and production processes. Evaluating alternative investments, equipment service life, product development, business plan development and marketing.
Prerequisites: none

SOCIAL: Social and Professional Issues
Introduces students to the social and professional issues that arise in the context of software engineering.
Prerequisites: none

Engineering Electives

ARCH: Software Architecture
Architectural design of complex software systems. Commonly-used software system structures, techniques for designing and implementing these structures, models and formal notations for characterizing and reasoning about architectures, tools for generating specific instances of an architecture, and case studies of actual system architectures. The role of standards, reuse, and quality. Skills needed to evaluate the architectures of existing systems and to design new systems in principled ways using well-founded architectural paradigms.
Prerequisites: SE5 or SENG 330 or SENG 365 or CSC 365

CBSE: Component-Based Software Engineering
Building large-scale and complex software systems from available parts by consistently increasing return on investment and time to market, while assuring high quality and reliability. The course covers advanced topics on software components and component-based software engineering from research and practice.
Prerequisites: SE5 or SENG 330 or SENG 365 or CSC 365

CSCW: Computer-Supported Collaborative Work
Most of the work that people do requires some degree of coordination and communication with others. Successful designs require: (1) social psychological insight into group processes; (2) computer science insight into mechanisms to organize information, coordinate, share, and communicate, and (3) HCI design insight to achieve successful designs for computer- mediated tools. The course focuses primarily on the first two and examines problems and solutions in group coordination and systems including group decision support, organizational memory, virtual spaces, and collaborative design.
Prerequisites: HCI

FTC: Fault Tolerant Computing
An introduction to selected issues in fault tolerant computing. Topics include: definitions of reliability, availability, safety, maintainability, testability and dependability; system protection through both hardware and information redundancy; quantitative methods for the evaluation of reliability; the design and test of integrated circuits; software fault tolerance and software testing. The course includes a number of case studies of practical fault tolerant systems.
Prerequisites: OSDC, DD

CG: Computer Graphics
Offers an introduction to computer graphics, which has become an increasingly important area within computer science. Computer graphics, particularly in association with the multimedia aspects of the World-Wide Web, have opened up exciting new possibilities for the design of human-computer interfaces. The purpose of this course is to investigate the principles, techniques, and tools that have enabled these advances.
Prerequisites: LA, ALG1

MMS: Multimedia Systems
The influence of technology, especially digital technology, on how we express ourselves, how we communicate with each other, and how we preceive, think about, and interact with our world. The invention and creative use of enabling technologies for understanding and expression by people and machines. Topics include: digital video representations; three-dimensional images; physical interfaces; computational tools and media that help people learn new things in new ways (tele-learning); knowledge representation; machine interpretation of sensory data.
Prerequisites: none

ALG2: Algorithms and Data Structures 2
This course covers algorithm design and analysis in software engineering. Specific topics include advanced data structures (such as Binomial heaps and Fibonacci Heaps), graph algorithms (such as minimum spanning trees, maximum flow, all-pairs shortest paths, and single-source shortest paths), and advanced design and analysis techniques (such as dynamic programming, greedy algorithms, linear programming, and amortized analysis).
Prerequisites: ALG1, DS2

ALG3: Analysis of Algorithms
General techniques for designing and analyzing algorithms; an in depth examination of several problems and algorithms with respect to their time and space requirements; advanced data structures; sorting and searching; graph algorithms; backtracking; NP-complete problems; approximation algorithms.
Prerequisites: ALG1, AFL, DS2

CC: Compiler Construction
Compilation, including: lexical analysis, syntax analysis, semantic routines, code optimization, block structured languages and interpreters. Students will implement a compiler-interpreter for a simple language.
Prerequisites: SE4, CFL, ALG1

CON: Concurrency
Introduction to the foundations of concurrency theory and the issues of specification and verification of concurrent systems. Topics include: Models of concurrency, such as Petri nets, labeled transition systems, and traces. Specification of concurrent systems/programs in formalisms including process algebras, statecharts, Petri nets and temporal logics. Verification techniques, such as bisimulation and model checking. Case studies involve coordination problems, controller design, communication protocols, hardware and user interface design.
Prerequisites: OSDC

PL: Programming Languages
The fundamental concepts of imperative, object-oriented, applicative, and logic programming languages.
Prerequisites: ALG1, CAS, SE4

AI: Artificial Intelligence
Philosophy of artificial intelligence. AI programs and languages, representations and descriptions, exploiting constraints. Rule based and heuristic systems. Applications to engineering.
Prerequisites: none

PATREC: Pattern Recognition
Parallel and sequential recognition methods. Bayesian decision procedures, perceptrons, statistical and syntactic approaches, recognition grammars. Feature extraction and selection, scene analysis, and optical character recognition.
Prerequisites: CTRL

ROBOT: Robotics
Structure and specification of robot manipulators. Homogenous transformations. Link description. Manipulator kinematics. Inverse manipulator kinematics. Velocity and static forces in manipulators. An introduction to manipulator dynamics. Linear control of robot motion. Model-based nonlinear control of robot manipulators.
Prerequisites: MECHSYS, CTRL

IKM: Information and Knowledge Management
Uses the idea of information as a unifying theme to investigate a range of issues in computer science, including database systems, artificial intelligence, human-computer interaction, multimedia system, and data communication.
Prerequisites: DB

NC: Network-Centric Computing
Presents those aspects of computer architecture that are central to communications and networking.
Prerequisites: OSDC, SE6

DC: Distributed Computing
Introduces concurrency in the context of distributed systems. The course covers both the abstract principles of concurrent programming and their concrete realization in distributed, network-based systems. Topics include the basic theory of concurrency, hardware and software features to support concurrency, concurrent and distributed algorithms, and middleware.
Prerequisites: OSDC, SE6

WMC: Wireless and Mobile Computing
This course focuses on the design and implementation of wireless and mobile computing *solutions*. The students study emerging technologies such as Jini, WAP, IEEE802.11, and Bluetooth. Targeted applications include handheld and mobile devices such as the Palm Connected Organizer, Handspring Visor, and PocketPC/WindowsCE devices such as the Compaq iPaq and HP Jornada.
Prerequisites: SAS

DSP: Digital Signal Processing
Generation of discrete-time signals through the sampling process and their spectral representation. Mathematical representation and properties of digital signal processing (DSP) systems. Typical DSP systems: digital filters and applications. The z-transform and its relation to the Laurent series. Evaluation of the inverse z-transform using complex series and contour integrals. Application of the z-transform for the representation and analysis of DSP systems. The processing of continuous-time signals using DSP systems. The discrete-Fourier transform and the use of fast Fourier transforms for its evaluation. Introduction to the design of DSP systems. Design project using Matlab.
Prerequisites: SAS

COM: Digital Communications
The course is concerned with the transmission, communication and processing of signals (information) and the necessary technical equipment for these purposes. Introduction to protocol engineering; PDU encoders and decoders; buffer management in communications programs; real-time constrains; timer management. Land and satellite-based mobile radio services, systems and networks. Mobile cellular telephone, paging, telepoint and wireless LAN systems. Switching and other protocols in support of mobility. Frequency reuse and channel allocation. Multiple access methods. Architectures of mobile distributed computing systems. Future developments in mobile telecommunications and associated new design problems. Communication via optical fibres.
Prerequisites: SAS, CTRL

NUM: Numerical Methods
The study of computational methods for solving problems in linear algebra, nonlinear equations, approximation, and ordinary differential equations. The student will write programs in a suitable high level language to solve problems in some of the areas listed above but the course will also teach the student how to use mathematical subroutine packages currently available in computer libraries.
Prerequisites: CALC2

NA1: Numerical Analysis I
An introduction to selected topics in Numerical Analysis. Typical areas covered: error analysis, roots of equations, systems of linear equations, linear programming, interpolation, numerical integration, and ordinary differential equations.
Prerequisites: SYSDYN

NA2: Numerical Analysis II
An introduction to selected topics in Numerical Analysis. Typical areas covered: ordinary differential equations, numerical differentiation, approximation of functions, iterative methods for linear equations, eigenvalues and eigenvectors, systems of nonlinear equations, boundary-value problems and partial differential equations.
Prerequisites: SYSDYN

ORLP: Operations Research: Linear Programming
An introduction to linear programming and its applications. Topics include: the simplex method, the revised simplex method, computer implementations, duality. Optional topics include: parametric and sensitivity analysis, primal-dual algorithm, network simplex method, the network flow problem, and game theory. Typical applications include: fitting curves to data, the transportation problem, inventory problems and blending problems.
Prerequisites: CALC2

ORSIM: Operations Research: Simulation
An introduction to discrete event simulation. Topics include: elementary queueing theory, basic techniques of discrete event simulation, generating random numbers, sampling from non-uniform distributions, simulation programming using general purpose languages and also special purpose simulation languages.
Prerequisites: CALC2

MIN: Data Mining
An introduction to data mining in the context of customer relationship management. Data preparation, model building, and data mining techniques such as clustering, decision trees and neural networks will be discussed and applied to case studies. Data-mining software tools will be reviewed and compared.
Prerequisites: DB

MIRCO: Microprocessor Systems
Introduction to microprocessor architecture. Instruction sets, addressing modes, and programming. Memories, I/O systems, and interfacing. Development systems. Application to engineering systems.
Prerequisites: CAS

CSA: Computer Systems and Architecture
Architecture and performance of modern processors, performance metrics; instruction set architectures and their impact on performance; instruction and arithmetic pipelines; pipeline hazards; exception handling; caches. Integral to the course is a Project Laboratory. Working in teams, students are expected to design and implement a processor based on a given specification of a simple instruction set. Student's progress is determined through a preliminary design review, a presentation, demonstration of the implementation and a final report.
Prerequisites: none

ERGO: Ergonomics
Accidents associated with "human error" often reflect the failure to recognize human factors in the design stage. This course reviews sensory, motor, and cognitive performance characteristics and derives human engineering design criteria. Principles of displays, controls and ergonomics are discussed.
Prerequisites: none

MCOM: Mobile Communications
Fading and shadowing, noise and interference effects; source coding, modulation, error control coding, spread spectrum and multiplexing techniques for mobile communications; capacity estimation and comparative (FDMA/TDMA/CDMA) analysis of PCN and Cellular Systems; capacity estimation for wireless PABX and LAN systems.
Prerequisites: SAS

AOOD: Advanced Object Oriented Design
Development and use of object-oriented design abstractions, with emphasis on the design of distributed object-oriented systems. Evaluation and selection of appropriate design patterns. Use of components. Distributed component models such as DCOM and CORBA. Use of models in the design of distributed object-oriented applications. Documentation standards such as UML.
Prerequisites: SE3

SP: Software Process
Software process design, modeling, implementation, management, assessment and improvement as well as other non-process factors that affect software quality. ISO 9001, SEIfs CMM. Group projects involving industry-relevant software process definition and assessment. Individual study of the research literature. ROI (Return on Investment) analysis.
Prerequisites: SE5

MSD: Management of Software Development
Non-functional requirements elicitation, configuration control, environments, product lines. Version control. Deployment. Time-to-market versus quality tradeoffs. Defect tracking.
Prerequisites: SE6

TOP: Topics in Software Engineering
The topics in this course depend primarily on the interests of the instructor. Entrance to the course will be restricted to third and fourth year students who meet the prerequisites specified for the topic to be offered. Some topics may require laboratory work as well as lectures.
Prerequisites: SE3, SE4