SIMON FRASER UNIVERSITY

S.77-95

MEMORANDUM

To: SENATE

From: D. R. BIRCH, CHAIRMAN
SENATE COMMITTEE ON UNDERGRADUATE STUDIES

Subject: FACULTY OF INTERDISCIPLINARY STUDIES - NEW COURSE PROPOSALS: CMPT 205-3, CMPT 293-3, CMPT 315-2

Date: SEPTEMBER 15, 1977

MOTION: "That Senate approve, and recommend approval to the Board of Governors, as set forth in Paper S.77-95, the new course proposals for

CMPT 205-3 - Introduction to Formal Topics in Computing Science
CMPT 293-3 - Introduction to Minicomputers and Microprocessors
CMPT 315-2 - Advanced Software Project."

Approval is given for CMPT 315-2 to be first offered in the Spring semester 78-1 and for CMPT 293-3 to be first offered in the Summer semester 78-2.

The above courses were considered by SCUS on July 12, 1977, with approval being given to the latter two courses and with request for further data on CMPT 205-3. This latter course was again considered with the additional data satisfactory to the Committee. The Senate Committee on Undergraduate Studies now recommends approval of the three courses.

SIMON FRASER UNIVERSITY

SCUS 7717

MEMORANDUM

To Mr. H. M. Evans, Registrar, and Secretary of the Senate Committee on Undergraduate Studies.	**From** J. Blanchet, Secretary, Faculty of Interdisciplinary Studies Undergraduate Curriculum Committee.
Subject CMPT 205-3, Introduction to Formal Topics in Computing Science.	**Date** July 27/77.

Attached is a revision of the above-named course which has been submitted to this office by the Computing Science Program. Would you please place this item on the Agenda of the Senate Committee on Undergraduate Studies for re-examination.

Thank you.

Janet Blanchet.

Attachment.

I.S.C. 77-21 (revised).

SENATE COMMITTEE ON UNDERGRADUATE STUDIES

NEW COURSE PROPOSAL FORM

1. **Calendar Information** Department: Computing Science

Abbreviation Code: CMPT **Course Number:** 205 **Credit Hours:** 3 **Vector:** 2-2-0

Title of Course: Introduction to Formal Topics in Computing Science

Calendar Description of Course: This course provides an introduction to the theoretical aspects of computing, building on computational concepts encountered in 103-3 and 105-3. Topics include discrete mathematical structures as they apply to computing science, and an introduction to the formal study of models of computation, formal languages and algorithms. This material is developed more extensively in subsequent upper level theory courses.

Nature of Course Lecture/Tutorial

Prerequisites (or special instructions):
CMPT 103-3 and CMPT 105-3

What course (courses), if any, is being dropped from the calendar if this course is approved: None

2. **Scheduling**

How frequently will the course be offered? Twice yearly

Semester in which the course will first be offered? 78-3

Which of your present faculty would be available to make the proposed offering possible? D. Kirkpatrick, R. Harrop, D. Granot

3. **Objectives of the Course**

SEE ATTACHED SHEET

4. **Budgetary and Space Requirements** (for information only)

What additional resources will be required in the following areas:

Faculty None

Staff ½ - 1 T.A. per offering

Library None

Audio Visual None

Space None

Equipment None

5. **Approval**

Date: July 25/77. 28 July 1977 13 Sept. 1977

Department Chairman

J. W. Calvert
Dean

Chairman, SCUS

SCUS 73-34b:- (When completing this form, for instructions see Memorandum SCUS 73-34a. Attach course outline).

(a) Course Objectives

This course provides an introduction to the formal study of basic computational concepts encountered at the 100 level. In varying depth, machines, languages, algorithms, data and information are all considered. In terms of mathematical preliminaries the course is self-contained. It is designed to motivate students with weak mathematical backgrounds. To this end, elementary concepts from discrete mathematics are introduced and applied in the context of practical computational problems.

Rationale:

(i) Computing Science is rapidly becoming a more formal discipline. This course introduces and motivates many of the formal tools that are used in advanced computing courses. At the same time this course should increase significantly the student's ability to critically assess the current literature in Computing Science.

(ii) Computing Science has an established yet expanding theoretical core. This course provides a springboard to the more intensive theoretical courses in our Program.

(b) Course Syllabus

I Discrete Mathematical Structures (4 weeks)
logical notation, expressions and reasoning. Sets, relations, functions, graphs and trees
- properties, computer representations and manipulations, applications
cardinality of sets, natural numbers, induction.

II Models of Computation (3 weeks)
nature of modeling, limitations, switching circuits, sequential machines, Turing machines, random access machines
- capabilities and limitations

III Languages (3 weeks)
languages in problem solving
programming and natural languages
- formal description, grammar, parsing
overview of compilation
- Polish notation, arithmetic and logical expressions

IV Algorithms and Programs (3 weeks)
design methodologies, structured considerations, efficiency concerns, analysis techniques, testing verification and proof of correctness

Suggested Textbook:

J.P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer Science (McGraw-Hill)

Alternative and supplementary sources:

F.P. Preparata and R.T. Yeh, Introduction to Discrete Structures for Computer Science and Engineering (Addison-Wesley)

D.F. Stanat and D.F. McAllister, Discrete Mathematics in Computer Science, (Prentice-Hall)

D.E. Knuth, The Art of Computer Programming, Vol 1., Fundamental Algorithms (Addison-Wesley)

A.V. Aho, J.E. Hopcroft and J.D. Ullman, The Design and Analysis of Computer Algorithms, (Addison-Wesley)

N. Wirth, Systematic Programming, (Prentice-Hall)

(c) Related Courses

The course takes CMPT 103-3 and 105-3 as prerequisites and will provide a foundation for such upper level courses as CMPT 305-3, CMPT 351-3, CMPT 400-3 and CMPT 410-4.

The proposed course provides an elementary treatment of some of the concepts encountered in MATH 306-3, MATH 401-3 and MATH 402-3. It should make these courses more accessible to all Computing Science students.

SIMON FRASER UNIVERSITY

SCUS 77-13

MEMORANDUM

Mr. H.M. Evans, Secretary, Senate Committee on Undergraduate Studies	From Janet Blanchet, Administrative Assistant Dean of Interdisciplinary Studies Office
Subject New Computing course proposals ISC77-21, ISC77-22 & ISC77-23	Date June 30, 1977

Attached are course proposals for:

ISC 77 - 21: Cmpt. 205 Introduction to Formal Topics in Computing Science,

ISC 77 - 22: Cmpt. 293 Introduction to Minicomputers and Microprocessors, and

ISC 77 - 23: Cmpt. 315 Advanced Software Project.

The above noted course proposals were approved by the Faculty of Interdisciplinary Studies Undergraduate Curriculum Committee on June 28th for furtherance to SCUS. Would it be possible to have these items placed on the agenda of the upcoming SCUS meeting?

J. Blanchet

JB:rmy
Attachments

I.S.C. 77-22

SENATE COMMITTEE ON UNDERGRADUATE STUDIES

NEW COURSE PROPOSAL FORM

1. **Calendar Information** **Department:** *Computing Science Program*

Abbreviation Code: *CMPT* **Course Number:** *293* **Credit Hours:** *3* **Vector:** *(0-1-3)*

Title of Course: *Introduction to Minicomputers and Microprocessors*

Calendar Description of Course:

SEE ATTACHED

Nature of Course *tutorial/laboratory*

Prerequisites (or special instructions):

CMPT 105 and a working knowledge of a high level programming language.

What course (courses), if any, is being dropped from the calendar if this course is approved: *None*

2. **Scheduling**

How frequently will the course be offered? *once per year (more if sufficient demand)*

Semester in which the course will first be offered? *summer 78*

Which of your present faculty would be available to make the proposed offering possible? *R. Hobson*

3. **Objectives of the Course**

Students should develop a sound, working knowledge of what can be done in a minicomputer/microprocessor environment. This is important because minicomputers and microprocessors contribute to a substantial portion of the professional computer scene. Participants will also be better prepared for the concepts which follow in upper division courses.

4. **Budgetary and Space Requirements** (for information only)

What additional resources will be required in the following areas:

Faculty *Laboratory Instructor*

Staff *none*

Library *none*

Audio Visual *none*

Space *none*

Equipment *students will be given the option of renting or buying a cassette and a floppy disk cartridge. If they elect to rent the charge should cover depreciation.*

5. **Approval**

Date: June 14/77.	June 30/77	13 Sept. 1977
V. Stel	J.W. Bolvert	D.R. Bruce
Department Chairman	Dean	Chairman, SCUS

SCUS 73-34b:- (When completing this form, for instructions see Memorandum SCUS 73-34a. Attach course outline).

Calendar Description

Introduction to Minicomputers and Microprocessors

CMPT 293-2 *(0-1-3)*

Students will gain hands-on practical experience in minicomputer and microprocessor environments. Inherent limitations and potentialities are emphasized. The course includes a hardware component orientation - CPU's, teletype, CRT, printer, magnetic tape, cassette, paper tape, moving head disk, floppy disk, and graphics devices. Operational topics include instruction sets, bootstrapping, stand-alone control, memory management, interrupts, debugging machine (assembler) code, slow speed device communication, high speed device communication, operating system control, and operating system generation.

Note: There will have to be several laboratory sessions available each week. One lab session will accommodate only about five students.

Introduction to Minicomputers and Microprocessors

Course Outline

Part I Basic Concepts (use Varian V75 to demonstrate)

Week I

Tutorial
- *Varian CPU design, registers, machine language instruction sets, introduction to assembler, device communication, implications of device speed vs CPU speed.*

Lab
- *Varian Hardware Orientation (various peripherals and CPU)*
- *Power up/down procedures*
- *stand-alone control, bootstrapping, introduction to I/O.*

Week II

Tutorial
- *communicating with slow speed devices: TTY, CRT, character printer, magnetic tape, cassette tape, papertape.*

Lab
- *debugging machine (assembler) code*
- *practical slow speed device communication problems.*

Week III

Tutorial
- *High speed communication (rotating disks)*
- *Direct Memory Access and I/O*

Lab
- *Disk communication*
- *other DMA applications (mag tape line printer etc.)*

Week IV

Tutorial
- *Operating System Control I (I/O drivers, logical devices)*
- *interrupts*
- *memory management*

Lab
- *interrupt handling*
- *bootstrapping the operating system*
- *basic operating system components (utilities)*

PROJECT ASSIGNED

Week V

Tutorial
- *Operating System Control II (Tasks, Software, System Generation)*
- *Software Conventions*
- *Spooling*

Lab
- *Text entry, compilation, load module generation*
- *debugging*

Part II Comparison of Varian V-75, PDP 11/34 and other minicomputer systems

Week VI

Tutorial
- *PDP 11 design and machine instruction set*
- *comparison of Varian and PDP 11 with some generalization*

Lab
- *familiarization with PDP 11/34*
- *communication with peripherals on the 11*

Week VII

Tutorial
- *Operating System Control*
- *comparison of operating systems*

Lab
- *user environment on the 11 (software, utilities etc.)*

Week VIII

Tutorial
- *special purpose devices (graphics, plotters)*

Lab
- *basic graphics I/O*

Part III Microprocessor Environments

Week IX

Tutorial
- *Intel 8080A MPU design, 8080 based kits and systems, machine instruction set and assembler*

Lab
- *familiarization with Mini-Micro designer*
- *ROM programs*

Week X

Tutorial
- *expanding a small system*
- *communication with terminals and a floppy disk*

Lab
- *practical communication problem*

Week XI

Tutorial
- *Motorola M6800 MPU design, 6800 based kits and systems, machine instruction set*
- *comparison of microcomputers*

Lab
- *practical familiarization with available M6800 kits*

Week XII

Tutorial
- *introduction to real time data acquisition and processing*

Lab
- *familiarization with some simple on-line experiments*

Week XIII

Tutorial
- *multi terminal systems*
- *networks*

Lab
- *finish off Varian project. (see Week IV)*

REFERENCE MATERIALS

The minicomputers and microprocessors have manuals supplied by their vendors. Course materials will be produced from appropriate sections of this documentation in the form of a booklet or booklets. A charge covering duplication costs will be levied.

I.S.C. 77-23

SENATE COMMITTEE ON UNDERGRADUATE STUDIES

NEW COURSE PROPOSAL FORM

1. **Calendar Information** **Department:** *Computing Science Program*

Abbreviation Code: *CMPT* **Course Number:** *315-2* **Credit Hours:** *2* **Vector:** _____

Title of Course: *Advanced Software Project*

Calendar Description of Course:

SEE ATTACHED

Nature of Course *group discussion (tutorial)*

Prerequisites (or special instructions):

- *prior written approval of the project supervisor*
- *CMPT 201 and other requirements as determined necessary by a project supervisor.*

What course (courses), if any, is being dropped from the calendar if this course is approved: *None*

2. **Scheduling**

How frequently will the course be offered? *every semester*

Semester in which the course will first be offered? *A.S.A.P.*

Which of your present faculty would be available to make the proposed offering possible? *all*

. **Objectives of the Course**

Many of our advanced courses provide students with a theoretical introduction to a topic. In some cases students will be motivated to seek further practical involvement with the subject. This course will enable the continuous development of systems which emanate as teaching materials from other upper division courses.

4. **Budgetary and Space Requirements** (for information only)

What additional resources will be required in the following areas:

Faculty *none*

Staff *none*

Library *none*

Audio Visual *none*

Space *none*

Equipment *none*

5. **Approval**

Date: June 14/77. June 30/77 13 Sept. 1977

T. C. Shrimpton — Department Chairman

J. W. Bolvert — Dean

D. R. Brink — Chairman, SCUS

SCUS 73-34b:- (When completing this form, for instructions see Memorandum SCUS 73-34a. Attach course outline).

Course Outline

Typical Ongoing Projects

I Information Organization and Retrieval

There are two prototype database systems which are functional according to limited specifications. Typical projects could involve extending the capabilities of these systems and improving upon chosen algorithms.

II Operating System Design

Our minicomputer and microprocessor environments are well suited for testing new operating system concepts. Students will become familiar with the development history of our operating systems. They will gain valuable experience by implementing new features according to recent research or by testing their own ideas.

III Interactive Graphics Facility

Projects will involve computer animation or a high level interactive graphics communication package.

IV Multiple Processor Systems

New LSI developments have brought the cost of parallel processing down to laboratory budget proportions. There are hardware and software design problems associated with parallel processing.

REFERENCE MATERIALS

Projects will normally be associated with other upper division courses. Reference materials will therefore be chosen from appropriate reading lists.

Calendar Description

CMPT 315-2 Advanced Software Project (Tutorial if necessary)

This course provides an opportunity for students to apply concepts expounded in other upper division courses. Ongoing projects in operating system design, information organization and retrieval, compiler construction, interactive graphics, firmware design, algorithm design, artificial intelligence, and numerical methods are available. These projects will normally involve group activity. Sponsorship for a given project depends upon the number of students interested and the availability of a faculty member.