RECEIVED
APR 11 1995
LIBERAL STUDIES

LSC Use Only
Number: _____
Submission Date: _____
Action-Date: _____

UWUCC USE Only
Number: ~~94-94~~ 95-15
Submission Date: Appr 9/26/95
Action-Date: Senate App 11/7/95

CURRICULUM PROPOSAL COVER SHEET
University-Wide Undergraduate Curriculum Committee

I. CONTACT

Contact Person James L. Wolfe Phone x6104

Department Computer Science

II. PROPOSAL TYPE (Check All Appropriate Lines)

XX COURSE Intro Systems Prgmng
Suggested 20 character title

____ New Course* _____
Course Number and Full Title

____ Course Revision _____
Course Number and Full Title

____ Liberal Studies Approval+ for new or existing course _____
Course Number and Full Title

____ Course Deletion _____
Course Number and Full Title

____ Number and/or Title Change _____
Old Number and/or Full Old Title

_____
New Number and/or Full New Title

XX Course or Catalog Description Change CO 430 Introduction to Systems Programming
Course Number and Full Title

_____ PROGRAM: _____ Major _____ Minor _____ Track

____ New Program* _____
Program Name

____ Program Revision* _____
Program Name

____ Program Deletion* _____
Program Name

____ Title Change _____
Old Program Name

_____
New Program Name

III. Approvals (signatures and date)

Department Curriculum Committee — Department Chair

College Curriculum Committee — College Dean

+Director of Liberal Studies (where applicable) — *Provost (where applicable)

Part II. Description of Curruculum Change

1. Complete Catalog Description

CO 430 Introduction to Systems Programming 3c-0l-3sh
Prerequisites: CO 300 and CO 310
Concepts and techniques of systems programming with an emphasis on assembly, linking, loading, and macro processing for user programs. Overviews of higher-level language translation and system control. Programming and research projects

2. Current (old) Catalog Description

CO 430 Introduction to Systems Programming 3c-0l-3sh
Prerequisites: CO 300 and CO 310
Concepts and techniques of systems programming with an emphasis on assembly and compilation of user programs. Representation of source language so as to facilitate the needed translation process. Exercises using various computer systems.

Revised Description

Concepts and techniques of systems programming with an emphasis on assembly, linking, loading, and macro processing for user programs. Overviews of higher-level language translation and system control. Programming and research projects

3. Rationale for the change.

The new description expands on the terminology used in translating and preparing user programs for execution at the lowest level of the system. The mention of higher-level language translation is deemphasized to reflect the amount of coverage of that topic. Mention is also made of "system control" - a topic that has always been part of the course. In essence, the new description better reflects and summarizes the actual course contents than the old description; the material being taught is not changing.

Part III. Letters of Support

None. This is a course for Computer Science majors.

IV. EVALUATION METHODS

The final grade for the course will be determined as follows:

50% Tests. At least two tests (mid term and final) consisting of mathematical problems and programming questions.

50% Homework and Projects.
At least six computer projects will be assigned. Projects will be graded on output and style. Short homework papers from the textbook problem sets may also be assigned.

V. REQUIRED TEXTBOOKS, SUPPLEMENTAL BOOKS AND READINGS

Textbook: Etter, D.M., Engineering Problem Solving with Matlab, Prentice Hall, 1993.

Supplemental Books: Metcalf, Michael and John Reid, Fortran 90 Explained, Oxford University Press, 1990.

VI. SPECIAL RESOURCE REQUIREMENTS

There are no special resource requirements for this course.

VII. BIBLIOGRAPHY

Burden, Richard and J. Douglas Faires, Numerical Analysis, Third Edition, Prindle, Weber & Schmidt, 1985.

Cheny, Ward and David Kincaid, Numerical Mathematics and Computing, Third Edition, Brooks/Cole, 1994.

Fortran 90, ISO/IEC JTC1/SC22/WGS Internal document N692 Submitted as Text for ISO/IEC 1539:1991, 1991.

Lastman, Gary J. and Naresh K. Sinha, Microcomputer-Based Numerical Methods for Science and Engineering, Saunders, 1989.

Mathews, John H., Numerical Methods for Mathematics, Science, and Engineering, Prentice Hall, 1992.

94-95

IUP Computer Science Department
CO250 Introduction to Numerical Methods

Course Syllabus

I. CATALOG DESCRIPTION

CO250 Introduction to Numerical Methods 3c-01-3sh

Prerequisite: CO110, MA122 or MA123 or MA127

Algorithmic methods for function evaluation, roots of equations, solutions to systems of linear equations, matrix operations, curve fitting, interpolation, numerical integration and differentiation; errors in computation.

II. COURSE OBJECTIVES

1. Students should know the role of and the limitations of the computer in solving mathematical and engineering problems.

2. Students should know how to implement mathematical algorithms using the Fortran 90 programming language.

3. Students should know how to use software packages such as Matlab as an aid in solving numerical problems.

4. Students should know selected numerical algorithms for solving a variety of commonly encountered mathematical problems.

III. COURSE OUTLINE

	Topic	Hours
1.	Introduction	
	a. Problem Solving	2
	b Matlab	2
	c Fortran 90	3
2.	Fundamental Computations	
	a. Number Representation and Errors	2
	b. Scalar and Array Computations	2
	c. Statistical Measurements	2
	d. Matrix Computations	2
	e. Plotting Capabilities in MatLab	2
3.	Numerical Algorithms	
	a. Locating roots of Equations	5
	b. Solutions to Systems of Linear Equations	6
	c. Interpolation and Curve Fitting	6
	d. Polynomial Analysis	6
	e. Numerical Integration and Differentiation	6