LSC Use Only No: 07-16a.	LSC Action-Date:	UWUCC USE Only No.	UWUCC Action-Date: APP-10/30/07	Senate Action Date: App-12/4/07

Curriculum Proposal Cover Sheet - University-Wide Undergraduate Curriculum Committee

Contact Person **James L. Wolfe**	Email Address **jlwolfe@iup.edu**
Proposing Department/Unit **Computer Science**	Phone **7-6104**

Check all appropriate lines and complete information as requested. Use a separate cover sheet for each course proposal and for each program proposal.

1. Course Proposals (check all that apply)

_____New Course __X__Course Prefix Change _____Course Deletion

__X__Course Revision _____Course Number and/or Title Change __X__Catalog Description Change

***Current* Course prefix, number and full title**	***Proposed* course prefix, number and full title, if changing**
COSC 250 Introduction to Numerical Methods	**COSC/MATH 250 Introduction to Numerical Methods**

2. Additional Course Designations: check if appropriate

_____ This course is also proposed as a Liberal Studies Course.

_____ This course is also proposed as an Honors College Course.

_____ Other: (e.g., Women's Studies, Pan-African)

3. Program Proposals

_____Catalog Description Change _____Program Revision

_____New Degree Program _____Program Title Change _____Other

_____New Minor Program _____New Track

***Current* program name**	***Proposed* program name, if changing**

4. Approvals		**Date**
Department Curriculum Committee Chair(s)		5 Dec 06
		12/8/06
Department Chair(s)		12/7/06
		12-7-06
College Curriculum Committee Chair		01/17/08
College Dean		9/24/07
Director of Liberal Studies *		
Director of Honors College *		
Provost *		
Additional signatures as appropriate: (include title)		
UWUCC Co-Chairs	Gail S. Sechrist	10/30/07

* where applicable

Received

SEP 25 2007

Liberal Studies

Part II. Description of Curriculum Change

1. New Syllabus of Record

I. Catalog Description

COSC 250 Introduction to Numerical Methods 3c-01-3cr

Prerequisite: COSC110, MATH121 or MATH125

Algorithmic methods for function evaluation, roots of equations, solutions to systems of linear equations, function interpolation, numerical differentiation and integration; and use spline functions for curve fitting. Focus on managing and measuring errors in computation. Also offered as MATH 250; either COSC 250 or MATH may be substituted for the other and may be used interchangeably for D or F repeats but may not be counted for duplicate credit.

MATH 250 Introduction to Numerical Methods 3c-01-3cr

Prerequisite: COSC110, MATH121 or MATH125

Algorithmic methods for function evaluation, roots of equations, solutions to systems of linear equations, function interpolation, numerical differentiation and integration; and use spline functions for curve fitting. Focus on managing and measuring errors in computation. Also offered as COSC 250; either COSC 250 or MATH may be substituted for the other and may be used interchangeably for D or F repeats but may not be counted for duplicate credit.

II. Course Outcomes

Upon completion of this course, students will be able to

1. Explain the role of and the limitations of the computer in solving mathematical and engineering problems.
2. Implement mathematical algorithms to
    a. evaluate functions
    b. find approximate roots of equations
    c. solve systems of linear equations
    d. perform numerical differentiation and integration
    e. fit a curve to a set of data.
3. Discuss selected numerical algorithms for solving a variety of commonly encountered mathematical problems.
4. Analyze a computation for error and discuss the types and sources of errors involved.
5. Explain how error accumulates and discuss the errors inherent in using standard floating point numbers.

III. Detailed Course Outline

A. Errors in Computation 0.5 week
    1. Representational error
    2. Computational error - relative and absolute
    3. Computer rounding approaches

B. Taylor Series representation of a function — 1.5 week
   1. Error term in the representation
   2. Properties of alternating series
   3. Appropriate and inappropriate applications

C. Representation of Numbers — 1.0 week
   1. Conversion of integers: binary - decimal - hex
   2. Conversion of floating point numbers: binary - decimal - hex
   3. Properties of the IEEE standard floating point representation
      a. Hole at zero
      b. Implied leading bit
      c. +/- infinity and Not-a-Number
      d. Machine epsilon
      e. Calculating roundoff error
      f. Propagated error
   4. Loss of significance
   5. Loss of precision theorem
   6. Techniques for avoiding loss of significance
      a. Rationalization
      b. Use of the Taylor series
      c. Use of identities

D. Finding Roots of Equations — 2 weeks
   1. Techniques
      a. Bisection method
      b. Newton's method
      c. Secant method
      d. Variation and hybrid methods
   2. Analysis of convergence for each technique
   3. Conditions under which convergence is a problem

E. Interpolation — 2 weeks
   1. Lagrange's form for the interpolating polynomial
   2. Newton's form for the interpolating polynomial
   3. Evaluation
   4. Divided differences algorithm
   5. Inverse interpolation
   6. Errors in interpolation
   7. Theorems regarding error
   8. Derivatives and divided differences
   9. Richardson extrapolation
   10. First and second derivatives using interpolation

F. Numerical Integration — 2.5 weeks
   1. Using upper and lower sums
   2. Trapezoidal rule
   3. Error analysis
   4. Recursive trapezoid formula
   5. Romberg algorithm
   6. Simpson's rule
   7. Gaussian quadrature
   8. Method of undetermined coefficients
   9. Legendre polynomials

G. Systems of equations — 2.0 weeks
   1. Gaussian elimination

2. Poorly conditioned matrices
3. Scaled partial pivoting
4. Tridiagonal systems
5. LU factorization
6. Iterative solution of linear equations
    a. Gauss-Seidel method
    b. Jacobi method

H. Splines for curve fitting — 1.5 weeks
1. First degree splines
2. Interpolation
3. Modulus of continuity
4. Second degree splines
5. Cubic splines

In-class examinations — 1 week

Total — **14 weeks**

Final Exam — During Final Exam Week

IV. Evaluation Methods

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

65% Examinations.
Three exams during the term and the final consisting of mathematical problems.

35% Projects and homework.
At least four computer projects will be assigned in addition to one or two sets of mathematical problems. The projects and problems will be similar to those provided in the text being used.

V. Example Grading Scale

90% - 100%	A
80% - 89%	B
70%-79%	C
60% - 69%	D
Below 60%	F

VI. Undergraduate Attendance Policy

Class attendance is regarded as being very important. Individual faculty may establish penalties for excessive numbers of unexcused absences. Excused absences will be allowed for illness, family emergencies, and involvement in university activities, such as sports. The penalties specified will meet university guidelines and be distributed to students with the course syllabus on the first day of class.

VII. Required Textbooks, Supplemental Books and Readings

Cheney & Kincaid, *Numerical Mathematics and Computing*, Fifth Edition, Brooks/Cole, 2004.

VIII. Special Resource Requirements

None.

IX. Bibliography

Collins, G.W., *Fundamental Numerical Methods and Data Analysis*, NASA Astrophysics Data Systems (online), 2003.

Chapra, S., *Applied Numerical Methods with MATLAB*, McGraw-Hill, 2005.

Chapra, S. & Canale, R., *Numerical Methods for Engineers*, McGraw-Hill, 2006.

Heath, M., *Scientific Computing*, McGraw-Hill, 2001.

Hoffman, J., *Numerical Methods for Engineers and Scientists*, CRC, 2001.

Mathews, J., *Numerical Methods for Mathematics, Science, and Engineering*, Prentice Hall, 1992.

Moler, C., *Numerical Computing using MATLAB*, Cambridge Press, 2004.

Overton, M., *Numerical Computing with IEEE Floating Point Arithmetic*, SIAM, 2001.

Press, W., Teukolsky, S., Vetterling, W., Flannery, B., *Numerical Recipes in C++*, 2nd edition, Cambridge Press, 2006.

2. Summary of proposed revisions

All references to the software tools used in the course have been removed from the catalog description and course syllabus. Also, the content is explained in greater detail.

3. Justification

The proposed cross-listing is a recognition that the material covered in numerical methods falls into an area where mathematics and computer science overlap. As such, Numerical Methods could reasonably be a part of either a mathematics or computer science curriculum. By cooperating between the departments, we believe that the course will attract additional students, because both departments intend to allow the course to be counted in their minors. There is not expected to be sufficient demand for both Mathematics and Computer Science to offer the course at the same time; the two departments will work out a sharing of the load by alternately offering the course or some other arrangement. No load changes will occur through such an arrangement because Mathematics will be proposing the cross-listing of MATH 219 as COSC 219; a trading of teaching load can occur when MATH 250 is offered.

As can be seen by comparing the two catalog descriptions, the topics covered in COSC/MATH 250 are not changing. However, the tools used must change. FORTRAN 90 has not been available at IUP since 2000 when the VAX VMS system was decommissioned. The last few times COSC 250 was taught, C++ was the language used and neither FORTRAN 90 nor MATLAB (mentioned in the syllabus) was employed. The new catalog description and associated syllabus of record do not mention the tools used in the course; instead, there is a concentration on the mathematical concepts. The choice of tools is left to the instructor.

The change in prerequisites is due to the Mathematics Department's change in the calculus sequence. MATH 125 is replacing MATH 123 and MATH 127 is no longer being taught. Consequently, the calculus prerequisite needed to be changed from MATH 121, 123 or 127 to MATH 121 or 125.

4. Old syllabus of Record

I. CATALOG DESCRIPTION

COSC250 Introduction to Numerical Methods 3c-01-3sh

Prerequisite: COSC110, MATH122 or MATH123 or MATH127

Algorithmic methods for function evaluation, roots of equations, solutions to systems of linear equations, interpolation, curve fitting, numerical differentiation and integration; errors in computation. Introduction to Fortran90 programming and introduction to the use of a mathematical software package to graph functions.

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

	Hours
Introduction to Fortran 90 and Mainframe Operating System	9
Program format and introduction	
Operating system commands, symbols, and logicals	
Using the editor and compile, link and run	
Real, Integer and derived data types	
Programs and procedures	
Modules	
Control Structures	
One Dimensional Arrays	
Input, Output and Files	
Errors in Computer Computations	3
Numerical calculations, precision, rounding	
Parameterized REAL variables in Fortran 90	
Conditioning and Stability	
Numerical algorithms and graphing	8-12
Data fitting by least squares approximation	
Introduction to MATLAB	
Bisection method for solving an equation	
Limitations of numerical algorithms	
Solving quadratic equations	
Newton's method for solving non-linear equations	
Secant method for solving non-linear equations	
Muller's method for solving non-linear equations	
Advanced topics in Fortran 90	5
Matrices and 2-dimensional arrays	

Array constructors for rank-n arrays	
Five classes of arrays	
Allocatable arrays	
Whole array operations, masks, sections	
More numerical algorithms	11-15
Solving systems of linear equations	
Solving a tridiagonal system of equations	
Interpolation	
Fitting a set of data points using a cubic spline	
Numerical differentiation	
Integration and numerical quadrature	
Hour exams	2

IV. EVALUATION METHODS

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

50 - 60% Tests.
At least three tests (two during the term and the final) consisting of mathematical problems and programming questions.

40 - 50% Projects, labs, quizzes, and homework.
At least five computer projects will be assigned. Projects will be graded on output and style. Quizzes, lab sessions and short homework papers from the textbook problem sets may also be assigned.

V. REQUIRED TEXTBOOKS, SUPPLEMENTAL BOOKS AND READINGS

Ellis, Phillips, and Lahey, *Fortran90 Programming*, Addison Wesley, 1994.

VI. SPECIAL RESOURCE REQUIREMENTS

There are no special resource requirements for this course. There is a Fortran90 compiler on the mainframe and MATLAB is available both in the Stright 220 lab and in the Johnson Data Center.

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.

Etter, *Engineering Problem Solving with MATLAB*, Prentice Hall, 1993.

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.

Kerrigan, *Migrating to Fortran90*, O'Reilly, 1993.

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

Metcalf, Michael and John Reid, *Fortran 90 Explained*, Oxford University Press, 1990.

Part III. Letters of Support

This course has primarily been a service course for Mathematics and Physics students, although a few other students from NS&M take the course. Letters of support from Physics and Chemistry are included below. Mathematics is expressing its support by signing the proposal.

Subject: Re: COSC/MATH 250
Date: Tue, 14 Nov 2006 11:21:06 -0500
From: "hershman" <hershman@iup.edu>
To: "Jim Wolfe" <jlwolfe@iup.edu>

Jim Wolfe
Computer Science Department
IUP

Dear Jim,

Since there is no substantive content change in the course as proposed, the Physics Department is happy to support the proposed changes in the syllabus of record.

Ken Hershman
Chairman
IUP Physics Department

----- Original Message -----
From: "Jim Wolfe" <jlwolfe@iup.edu>
To: <hershman@iup.edu>
Sent: Monday, November 13, 2006 4:18 PM
Subject: COSC/MATH 250

> Ken,
>
> I don't think you responded to my request for a letter of support for
> the revision to COSC 250. Attached is a slight change in it. We are
> now proposing the same content as before but we are proposing the course
> be cross listed with Math. The intent is to then allow COSC 250 to be
> counted in the Computer Science minor (it counts that way now) or to
> allow MATH 250 to be counted in the Math minor. You can read more in
> the justification section.
>
> I am requesting your letter of support for this proposal.
>
> Jim

Subject:
Date: Thu, 12 Oct 2006 13:29:41 -0400
From: "Wendy L Elcesser" <endyw@iup.edu>
To: jlwolfe@iup.edu
CC: akondo, ssowa, mbriggs, woolcock

Dear Jim,

The Department of Chemistry Curriculum Committee was forwarded your request for a letter of support from the Chemistry Department for the revisions of COSC 250, Numerical Methods, and COSC 300, Computer Organization & Assembly Language. Although neither of these courses are actually required of our students, we appreciate that you have kept us informed of these upcoming revisions. Some of our students do take these courses as part of a potential minor in Computer Science. We also have had some of our students double major with Computer Science or choose Computer Science as the complementary field for the BA Chemistry degree program.

We have no objections to the changes proposed for these courses.

Thanks for reading.

Wendy Lou Elcesser
Chemistry Curriculum Committee Chairperson