Prefix, Number and Name of Course: MAT 366 Computer Mathematics II

Credit Hours: 3

In Class Instructional Hours: 3 **Labs: 0** **Field Work: 0**

Catalog Description:

Prerequisites MAT 164, MAT 241 and MAT 270 or instructor's permission.
Structured programming, verification of program validity, data structures, combinatorial problems, flow network algorithms, random number generators, simulation of random and nonrandom processes.

Reasons for Revision:

This is a revision of MAT 366. The revision is done in light of the Committee on Undergraduate Programs in Mathematics guidelines, and was necessary to update the literature and language used to describe the course, as well as to fit better with MAT 241 (Computer Mathematics I). This course will

1. provide mathematicians and scientists with the necessary knowledge to use computers effectively in their fields
2. provide an in-depth introduction to computer organization for mathematicians and scientists
3. provide an overview of computer languages and the mathematics behind them
4. provide further mathematics applications to computer science.

Student Learning Outcomes:	Course Content References:	Assessment:
1. Students will be able to write computer programs to do mathematical modeling	I. D-E II. III. IV.	1. Group work in class, individual homework assignments, projects, exams
2. Students will be able to describe computer programming organization and practices	I. A-F	2. Group work in class, individual homework assignments, projects, exams
3. Students will be able to determine the feasibility of solving a problem computationally	V. A-E	3. Group work in class, individual homework assignments, projects, exams
4. Students will be able to write and analyze algorithms.	IV. V.	4. Group work in class, individual homework assignments, projects, exams
5. Students will be able to use recursive functions effectively to create computational mathematical models	III.	5. Group work in class, individual homework assignments, projects, exams

Course Content:

I. Elements of Program Design
   - A. Overview of Programming Languages
   - B. Influences on Programming Language Design
     1. Mathematical influences
     2. Other influences
   - C. Procedural and Declarative languages
   - D. Overview of programming paradigms
     1. Structured programming
     2. Object Oriented programming
   - E. Program documentation
   - F. Informal Semantics of programming languages
     1. Pseudo-code
     2. The contour model

II. Data structures with mathematical applications
   - A. Arrays, files and sets
   - B. Linked lists
   - C. Stacks and queues
   - D. Binary trees
   - E. Heaps
   - F. Graphs, digraphs, networks and their implementation

III. Recursion
   - A. The substitution method
   - B. The recursion-tree method
   - C. The master method
   - D. Proofs of methods
   - E. Mathematical induction and recursion

IV. Sorting and searching algorithms
   - A. Heapsort
   - B. Quicksort
   - C. Sorting in linear time
   - D. Divide and conquer: Mergesort
   - E. Bitonic sort

V. Algorithm analysis and problem complexity
   - A. Probabilistic analysis and randomized algorithms
   - B. Growth of functions: Asymptotic notation
   - C. Algorithm complexity
   - D. Problem complexity
   - E. Introduction to np-completeness

Resources:

Classic Scholarship in the Field:

Aho, A.V. Ullman, J.D. and Hopcroft, J.E. (1983). Data Structures and Algorithms, Addison Wesley.
Cipra, B. (1993). What's Happening in the Mathematical Sciences (V1). AMS.
Cipra, B. (1994). What's Happening in the Mathematical Sciences (V2). AMS.
Cipra, B. (1996). What's Happening in the Mathematical Sciences (V3). AMS.
Cipra, B. (1999). What's Happening in the Mathematical Sciences (V4). AMS.
Garey, M. and Johnson D. (1979). Computers and Intractability. W.H. Freeman and Company.
Maeder, R.E. (1996). The Mathematica Programmer II. Academic Press.
Michaels, J.G. and Rosen, K.H. (1991). Applications of Discrete Mathematics. McGraw Hill.
Okasaki, C. (1999). Purely Functional Data Structures. Cambridge University Press.
Smith, J.T. (1999). C++ Toolkit for Engineers and Scientists, Thomson Computer Press.
Stanton, D. and White, D. (1986). Constructive Combinatorics, Springer-Verlag.
Weiss, M.A. (1998). Data Structures and Algorithm Analysis in C++ (2nd Ed). Addison Wesley.

Current Scholarship in the Field:

Bender, E.A. and Williamson, S.G. (2005). Mathematics for Algorithm and Systems Analysis.
Borwein, J., and Bailey, D. (2004). Mathematics by Experiment: Plausible Reasoning in the 21st Century. A.K. Peters.
Borwein, J., Bailey, D. and Girgensohn, R. (2004). Experimentation in Mathematics: Computational Paths to Discovery. A.K. Peters.
Cormen, T.H. Leiserson, C.E. Rivest, R.L. and Stein, C. (2001). Introduction to Algorithms, Second Edition. Prentice Hall.
Deitel, H. and Deitel, P. (2004). Java: How to Program (6th Ed). Prentice Hall.
Deitel, H. and Deitel, P. (2005). C++: How to Program (5th Ed). Prentice Hall.
Dewhurst, S. (2005). C++ Common Knowledge: Essential Intermediate Programming. Addison-Wesley Professional.
Drozdek, A. (2004). Data Structures and Algorithms in C++ (3rd Ed). Course Technology.
Halvorson, M. (2003) Microsoft Visual Basic .NET Step by Step. Microsoft Press.
Harbour, J.S. (2002). Microsoft Visual Basic .NET Programming for the Absolute Beginner. Course Technology PTR.
Jones, B.L. and Liberty, J. (2004). Sams Teach Yourself C++ in 21 Days. Sams.
Keogh, J. and Davidson, K. (2004). Data Structures Demystified. McGraw-Hill Osborne Media.
Lafore, R. (2002). Data Structures and Algorithms in Java (2nd Ed). Sams.
McMillan, M. (2005). Data Structures and Algorithms Using Visual Basic.NET. Cambridge University Press.
Meyer, S. (2005). Effective C++: 55 Specific Ways to Improve Your Programs and Designs. Addison-Wesley Professional.
Oualline, S. (2003). Practical C++ Programming, Second Edition (2nd Ed). O'Reilly Media, Inc.
Ruskeepaa, H. (2004). Mathematica Navigator: Mathematics, Statistics, and Graphics, Second Edition. McGraw-Hill Osborne Media.
Stroustrup, B. (2000). The C++ Programming Language. (3rd Ed). Addison-Wesley Professional.
Weiss, M.A. (2005). Data Structures and Problem Solving Using Java (3rd Ed). Addison Wesley.

Periodicals:

College Mathematics Journal
Discrete Mathematics and Computer Science Journal
Mathematics Magazine
The American Mathematical Monthly
The Journal of Undergraduate Mathematics and Its Applications (COMAP)

Electronic and/or Audiovisual Resources:

The On-Line Encyclopedia of Integer Sequences
(http://www.research.att.com/~njas/sequences/)
The electronic book A=B by Marko Petkovsek, Herbert Wilf and Doron Zeilberger
(http://www.cis.upenn.edu/~wilf/AeqB.html)
IEEE Annals of the History of Computing
(http://www.computer.org/portal/site/annals/)
The History of Computing Projects
(http://www.thocp.net/)
Decode Systems
(http://www.decodesystems.com/)
C/C++ Tutorial
(http://cplus.about.com/od/beginnerctutorial/l/blcplustut.htm)
Online C++ Tutorial
(http://www.intap.net/~drw/cpp/0