Prefix, Number and Name of Course: MAT 241 Computer Mathematics I

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

Catalog Description:
Prerequisites MAT 161 and MAT 163 or equivalent, or instructor's permission.
Fundamental concepts of problem solving by computer as applied to mathematics. Computer organization, operations and functions, algorithm development, programming techniques. Numerical methods as used in calculus, linear algebra, geometry, etc. Will use a computer language to be applied in this and other mathematics classes.

Reasons for Revision:
This is a revision of MAT 141. The course number was changed so that the new prerequisites have a lower course number. The content revision was necessary to reflect changes in the field of Mathematics, the associated programming languages and also in the BSC Mathematics Department. This revised course will
1. provide students with an introduction to computational tools available to mathematicians.
2. provide students with the background to write computer programs to explore mathematics.
3. provide students with the background to write computer programs to model mathematics and science.
4. teach the mathematics necessary to understand the modern computer organization.

Student Learning Outcomes:	Course Content References:	Assessment:
1. Students will be able to describe computer organization.	I A	1. Group work in class, individual homework assignments, projects, exams
2. Students will be able to describe different programming paradigms, including structure programming and object oriented programming.	I A-B	2. Group work in class, individual homework assignments, projects, exams
3. Students will be able to write algorithms and algorithm analysis.	I B	3. Group work in class, individual homework assignments, projects, exams
4. Students will be able to write simple computer programs that are well documented.	I B	4. Group work in class, individual homework assignments, projects, exams
5. Students will be able to use binary, octal and hexadecimal number systems.	I A 1.	5. Group work in class, individual homework assignments, projects, exams
6. Students will be able to understand, construct and use algorithms to solve mathematical problems.	II. A - E	6. Group work in class, individual homework assignments, projects, exams
7. Students will use computer generated data to explore mathematics.	II A-E	7. Group work in class, individual homework assignments, projects, exams

Course Content:

I. Computational mathematics and programming
    A. Introduction

        1. Computer arithmetic

        2. Binary, octal, and hexadecimal number systems

        3. Representation of negative numbers: 1's and 2's complement

        4. Real number representation

        5. Logical circuits and Boolean algebra

    B. Elements of programming languages

        1. Data types: integers, real numbers, arrays

        2. Operations: arithmetic, logic, precedence

        3. Statements: I/O, assignment, branching, loops

        4. Procedures and functions

II. Applications

    A. Topics from number theory

        1. Primarity test

        2. GCD, LCM, Euclidean algorithm

        3. Factoring a number

        4. Fibonacci sequence (non-recursive)

    B. Topics from statistics

        1. Finding the mean

        2. Finding the median

        3. Finding minimum and maximum

    C. Topics from geometry

        1. Work with coordinates

        2. Drawing an n-gon

        3. Drawing parametric curves: sine, cosine, and others

        4. Drawing a circle, ellipse

        5. Drawing quadratic curves

    D. Topics from linear algebra

        1. Addition and subtraction of matrices

        2. Multiplication of matrix by a scalar

        3. Multiplication of matrices

        4. Transpose of a matrix

        5. Evaluation of determinants

        6. Solution of a system of equations

    E. Topics from calculus

        1. Finding the roots of a linear equation and quadratic equation

2. Value of a polynomial

3. Root finding by halving the interval

4. Simple integration (Simpson's method, trapezoid method)

Resources:

Classic Scholarship in the Field:

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.
Smith, J.T. (1999). C++ Toolkit for Engineers and Scientists, Thomson Computer Press.
Stanton, D. and White, D. (1986). Constructive Combinatorics, Springer-Verlag.

Current Scholarship in the Field:

Borwein, J., Bailey, D. and Girgensohn, R. (2004). Experimentation in Mathematics: Computational Paths to Discovery. A.K. Peters.
Borwein, J. and Bailey, D. (2004). Mathematics by Experiment: Plausible Reasoning in the 21st Century. A.K. Peters.
Deitel, H. and Deitel, P. (2005). C++: How to Program (5th Ed). Prentice Hall.
Deitel, H. and Deitel, P. (2004). Java: How to Program (6th Ed). Prentice Hall.
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.
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.
Stroustrup, B. (2000). The C++ Programming Language. (3rd Ed). Addison-Wesley Professional.

Periodicals:

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

Electronic or Audiovisual Resources:

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/)
Java Tutorial
        (http://www.ibiblio.org/javafaq/javatutorial.html)
Electrical Engineering Training Series. Chapter 1. Number Systems
        (http://www.tpub.com/content/neets/14185/)