Course Description
Fundamental algorithms and processes used in business information systems. Development and representation of programming logic. Introduction to implementation using a high-level programming language.

Prerequisites
CSCE 101 or MGSC 290

Goals and Learning Outcomes:
1. Create pseudocode that describes the intended operation and flow of an application.
2. Specify and apply appropriate data types used in an application.
3. Develop correct branching and looping constructs needed to perform specific computing processes.
4. Create an appropriate user interface for an application.
5. Debug a medium complexity software application, identifying logic and syntax errors and apply appropriate changes to the code.
6. Develop a non-trivial Visual Basic program.

Course Text and Readings

Overall Structure of the Course

Course Requirements

Course Policies
Students should be familiar with and abide by the Code of Student Academic Responsibility. All violations will be regarded as serious and will result in a minimum penalty of failure in the course. Incompletes will be given only when serious and unanticipated circumstances prevent the timely completion of course work.

Assessment and Grading
30% In-class quizzes and exercises (Lowest 2 dropped)
40% Programming Assignments (None dropped)
30% Midterm (3) and Final Exams (Lowest 1 of 4 dropped)
Cutoffs no higher than the following will be used:
90%--A  80%--B  70%--C  60%--D
Course Outline

Topics to be covered, not necessarily in this order:

- Understanding of, and ability to generate pseudocode.
- Best practices in the creation of application
  - Object/Variable naming (i.e., adding a prefix to object and variable names that identifies data types, camel case, etc.)
  - Use of comments
  - Encapsulation
  - Variable scope
- Development of a user interface
  - Forms, Buttons, labels, textboxes, checkboxes, radio buttons, panels, groupbox
  - Be able to set the basic properties on those objects.
- Expressions
  - Addition, numeric data
- Non-numeric data, Strings
- Coding logic
  - Assignment statement
  - Branching statements
  - Looping statements
  - Executing code on startup
    - Form_Load event
- Exception Handling
  - Try … Catch
- Debugging skills
  - Break points
  - Auto and Local window
  - Diagnostic output
  - Watch variables
- Creating subroutines and functions
- Reading and Writing files
- Working with multiple forms