Course Outline for CS 35

ADVANCED VISUAL BASIC PROGRAMMING

Effective: Fall

I. CATALOG DESCRIPTION:
   CS 35 — ADVANCED VISUAL BASIC PROGRAMMING — 4.00 units

   This is an advanced course in Visual Basic programming. The contents will build on the skills acquired in earlier courses. Topics include advanced OOP, class libraries, Input and Output, graphics, files, multimedia, database, prototyping, interface design.

   3.00 Units Lecture 1.00 Units Lab

   Prerequisite
   CS 32 - Visual Basic Programming
   with a minimum grade of C

   Grading Methods:
   Letter or P/NP

   Discipline:

	MIN
Lecture Hours:	54.00
Lab Hours:	54.00
Total Hours:	108.00

II. NUMBER OF TIMES COURSE MAY BE TAKEN FOR CREDIT: 1

III. PREREQUISITE AND/OR ADVISORY SKILLS:
   Before entering the course a student should be able to:

   A. CS32
      1. GENERIC: These outcomes are being developed throughout the entire programming sequence. Upon completion of the course, to an intermediate level, students should be able to: Programming Skills
      2. present the elements and features of the development environment;
      3. explain and use the design process;
      4. define and use functions;
      5. define and explain trends in programming standards;
      6. write, compile, test and debug programs;
      7. present the characteristics of object-oriented programming;
      8. define and use data types and variables;
      9. define and use multi-dimensional arrays;
      10. define and use user interfaces;
      11. define and use file I/O;
      12. define and develop class modules;
      13. develop and use event-driven programs;
      14. Systems Analysis
      15. develop high-level systems and functional specifications;
      16. define general scope of work to meet requirements and constraints;
      17. Systems Design
      18. specify major subsystems and interfaces;
      19. develop detail design specifications;
      20. select design methodology and tools;
      21. identify maintenance requirements;
      22. Technical Documentation
      23. write in a concise and precise form appropriate for technical documentation;
      24. explain and use the processes and techniques of technical documentation;
      25. record system specifications accurately and completely;
      26. Testing and Debugging
      27. select debugging and testing methodology, and develop comprehensive and systematic test plan;
      28. develop testing procedures;
      29. conduct tests in the most efficient way;
      30. test programs, and document errors and solutions;
      31. User Interface Design

32. define the requirements for the user interface;
33. detail the development process and methods best suited for the project;
34. develop user interface (UI) to meet user requirements;
35. test Uis;
36. Problem Solving
37. recognize a wide range of problems, and assess their impact on the system;
38. use a wide range of troubleshooting methods and tools to isolate problems;
39. select the appropriate approach to identify causes of the problem based on the given situation;
40. SPECIFIC: These outcomes are detailed specifically for this course. Upon completion of the course students should be able to: write programs using the basic grammar and syntax of Visual Basic.
41. write programs using OOP concepts.
42. write programs using VBScript in web applications.
43. write programs using VBA and OLE.
44. write programs using advanced language capabilities: ActiveX, multimedia, timer, mouse controls, and graphical images.
45. write programs using specialized capabilities (such as User Defined structures, SQL and Multiple Document Interface structures).

IV. MEASURABLE OBJECTIVES:
Upon completion of this course, the student should be able to:

A. GENERIC: These outcomes are being developed throughout the entire programming sequence. Upon completion of the course, to an advanced level, students should be able to: Programming Skills
1. Explain and use the design process
2. Define and use functions and storage classes
3. Define and explain trends in programming standards
4. Write, compile, test and debug programs
5. Present the characteristics of object-oriented programming
6. Define and use data types and variables
7. Define and use multi-dimensional arrays
8. Define and use user interfaces
9. Define and use file I/O
10. Define and develop class modules
11. Develop and use event-driven programs
12. Embed one language in another

B. Database Design
1. Explain database design concepts and the role of database components
2. Create and customize forms and reports
3. Explain the use of databases and information in the business environment
4. Develop database business applications

C. Testing and Debugging
1. Select debugging and testing methodology, and develop comprehensive and systematic test plan
2. Design testing programs to uncover hardware compatibility problems during the development phase of the project
3. Develop testing procedures
4. Conduct tests in the most efficient way
5. Test programs, and document errors and solutions

D. User Interface Design
1. Define the requirements for the user interface
2. Develop and test prototypes
3. Construct user interfaces for flexibility and adaptability

E. Problem Solving
1. Recognize a wide range of problems, and assess their impact on the system
2. Use a wide range of troubleshooting methods and tools to isolate problems

F. Project Management
1. Present the main steps and issues in project management
2. Evaluate project requirements, and clearly define and articulate project scope and goals
3. Identify critical milestones and project performance, budgets and the use of resources

G. Task Management
1. Break down projects and activities into a series of tasks

H. SPECIFIC: These outcomes are detailed specifically for this course. Upon completion of the course students should be able to: Write programs that involve advanced OOP.
I. Write programs that use Class libraries.
J. Write programs using file I/O.
K. Write programs using Graphics.
L. Write programs using Multimedia.
M. Write programs using a database.
N. Write programs using prototyping.
O. Write programs using advanced interface design.

V. CONTENT:
A. Advanced OOP
B. Class libraries
C. File I/O
D. Graphics
E. Multimedia
F. Database
G. Prototyping
H. Advanced Interface design

VI. METHODS OF INSTRUCTION:
A. **Lecture** -
B. **Lab** - Lab Programming Assignments
C. **Discussion** -
D. **Demonstration** -
E. **Projects** - Optional: Programming project completed in teams

VII. TYPICAL ASSIGNMENTS:
A. Write a Windows GUI application to place a user designed graphic on the screen. 1. An example might be to have the students design a television using the Shapes Components 2. Apply buttons, a mainmenu, or other appropriate components

to the television to simulate use. B. Write a GUI program that asks a workers age and years of service and determines whether that worker is eligible for retirement based on the following rules: 1. All employs are eligible to retire at 65. 2. All employees are eligible to retire after 35 years service. 3. At 60, employees may retire with 25 years service. 4. At 55, employees may retire with 30 years service. 5. Input comes from data components: checkboxes, radiogroups, editboxes, etc. C. An optional team project to write a program to create a WEB browser with an attached database which tracks users, passwords, and web sites visited.

VIII. EVALUATION:
A. **Methods**

B. **Frequency**

1. Frequency of evaluation
   a. Recommend 2 or 3 exams plus final examination
   b. Recommend programming assignment to cover each topic within course content. Contents can be combined.
2. Types of Exam Questions
   a. Write a function called load() to read records from a binary data file and place the data into a listbox. Provide appropriate spacing for header control.
   b. Write a program to concatenate two text files: that is, to produce a file that contains the first file followed by the second file.
   c. Using the database methods described in class write the pseudocode to demonstrate how to use the methods to produce a report shown on page 321 of the text.

IX. TYPICAL TEXTS:
1. Julia Bradley and Anita Millspaugh *Programming in Visual Basic 6.0.*, Irwin-McGraw-Hill, 1999.
2. Edward J. Coburn *Programming with Visual Basic 6.0.*, Brooks/Cole, 2000.
3. Kip Irvine & Kaiyang Liang *Advanced Visual Basic 6* . 2nd ed., Scott Jones Publishing, 1999.

X. OTHER MATERIALS REQUIRED OF STUDENTS:
A. Current version of Microsoft Visual Basic
B. Students must have access to computers capable of compiling in the chosen language.