Q 'o(


***CIS 150: Introduction to Programming***

# Syllabus

## Course Description

Through a combination of lectures, lab activities, and project assignments, this course will cover the basic concepts and techniques of programming and algorithm development. Visual Basic will be used as the programming language. The course is the first of a series of core courses required for a major or a minor in *Computer Information Systems* at Chaminade University. Although there are no specific programming prerequisites for the course, a familiarity with the use of the microcomputer using the Windows environment is assumed.

## Course Objectives

The course is intended to help the students to:

- Become acquainted with the basic concepts of the computer, its organization, and software systems
- Become familiar with the top-down, Stepwise refinement approach to algorithm design
- Become familiar with the modular approach to program development
- Learn a collection of basic algorithms
- Become acquainted with event-driven programs
- Learn the basic features of the Visual Basic language
- Develop basic problem-solving skills using the computer
- Lay a basis for subsequent courses in Computer Information Systems

## Text Book

*Essentials of Visual Basic 6. 0 Programming,* by David Schneider. Prentice Hall, 1999. ISBN 0-13-012720-5

*Return to Top*

## Requirements

The following is a summary of what is expected of you for the course. Refer to the section on Grading for further details.

- Class Attendance
- Readings and Exercises
- Project Assignments
- Quizzes (unscheduled)
- Tests
- Final Exam

Class attendance is important, since main points of the course will be highlighted and details and examples will be demonstrated in the class. Readings will provide you with further explanations on the concepts and techniques covered in the class. Exercises will help you to reinforce key ideas covered in the class and to prepare you for tests. (Many of the test questions will be based on such Exercise questions). Since you learn by doing, *project assignments will* be the most important element among your responsibilities in the course.

*Return to Top*

## Project Assignments

Ten project assignments are scheduled for the semester. In general, you will have one week to complete an assignment. The schedule for project assignments are indicated in the class Schedule. Get in the habit of starting early on your assignments. Late assignments will be assessed a penalty which is equivalent to one letter grade per day. Regular credits will not be given for a program that is past its due date by one week or more. (However, a minimum value will be entered in the database for assignments that were turned in very late to distinguish them from those that were not turned in at all.) The procedure for submitting assignments will be announced in the class.

If you have difficulty with your programs, get help immediately so that you can stay on top of your assignments. When you are seeking help from the instructor in debugging your programs, always accompany your questions with a hardcopy of your program listing or a copy of your algorithm written in pseudocode.

You are expected to spend extra time beyond the class time on your project assignments. Check for posted computer lab hours. Be conscientious in completing your assignments, since they are indispensable to learning algorithm development and programming.

*Return to Top*

## Tests

Four tests, in addition to the final exam, are scheduled for the semester. Their dates are indicated in the course Schedule. These tests will cover concepts and programs covered in the class and in project assignments. In general, there will be no make-up quizzes and tests. Special cases will be considered when there are valid reasons, but arrangements must be made *before* the scheduled test dates.

*Return to Tot?*

## Submitting One's Own Work

Each student is expected to write his or her own programs that are to be submitted as project assignments. Although modern programming practices require

**extensive teamwork, one of the main goals in this class is that each student learn the basic programming skills by practicing individually. You must distinguish between consulting your friends or discussing problems with them from copying other people's work. Even if you "work together," each work must be different from another. The penalty for copying in tests, quizzes, and project assignments is, for the first offense, a grade of 0 for all parties involved; for the second offense, an F for the course.**

# Attendance

**Regular class attendance is important since you are responsible for all materials covered in the class. Attendance will be taken at all class sessions. Attendance will be considered as part of your final grade. If you need to be absent from a class, you should let the instructor know so that he can help you in catching up. Generally speaking, there will be no make-up tests. Make-up tests will be considered, with prior arrangements, only for excused absences because of serious reasons. Be sure to inform the instructor when you foresee that you must miss a test. A missed test due to an unexcused absence receives a grade of 0.**

*Return to Top*

# Grading

**The semester grade will be based on the following elements of your course responsibilities.**

```
Tests:                          60 x (4) = 240
Project Assignments:            25 x (10) = 250*
Quizzes:                        10 x (6) = 60**
Class Participation (attendance):      50
Final Exam:                           100
-----------------------------------------
Total:                                700**
```

* A **minimum of 7 completed assignments is a necessary condition for a passing grade.**

** **The total is subject to change depending on the number of quizzes.**

**The following guidelines will be used in determining the final grades.**

```
A: >=90; B: >= 80; C: >= 70; D: >= 60; F: < 60
```

**Test dates are indicated in the Course Schedule.**

*Return to Top*

# Criteria for Project Assignments

**Generally speaking, for each programming assignment submit:**

- **hardcopy of the source code**

- **image of the program interface**

**An exception is PA No. 1, which requires only the interface. How to produce the hard copy and an image of the interface will be discussed in the class. (You can also refer to the How To section.) If there are more than one problem involved, submit a hardcopy and a screen image for each program. The grade for each assignment will be based on the correctness and completeness of both the interface and the code.**

**Your projects will be evaluated on the following points.**

- **Correctness--does it work as advertised?**
- **Completeness--does it satisfy all requirements in the problem?**
- **Understandability--is the code easy to understand and is written in the simplest way possible?**
- **Interface--is the interface orderly, attractive, and easy to understand?**

**Check the following points in your code.**

- **Include documentation--name, PA ID, etc.**
- **Include summary description of program.**
- **Insert the** *Option Explicit* **statement (declare all variables).**
- **Indent the body of subprograms and functions.**
- **Indent the body f decision and loop structures.**
- **Insert a comment before a major section in the code and a blank line after it.**
- **Insert a space before and after each operator; e.g.,** `c = a + b, not c=a+b.`
- **Begin each variable name in lowercase; each procedure name in uppercase.**
- **Each function and subprogram should be logically simple; e.g., a procedure that inputs data should not also calculate.**
- **All interface objects should have user-defined names instead of default names--e.g.,** txtName.

*Return to Top*

## Getting Help

**For "quick" questions an email is the simplest way to contact me. Feel free to drop in at my office (Kieffer Hall, HH) during office hours or to set up an appointments outside those hours. (I am around my office usually in the afternoon.) When you need help in** debugging programs, bring a copy of the code **that is causing the problem.**

*Return to Top*

# CIS 150 Class Schedule

Chaminade University of Honolulu

Syllabus Schedule Resources Office Hrs Home

| Wk | Date, | Topics | Readings | Exercises | Due |
|---|---|---|---|---|---|
| 1 | | I. Introduction | 1.4 Intro to VB (p22-25) | | |
| | | • Preliminaries<br>o Pretest<br>o Syllabus<br>o Terminology<br>o Local System | 15. Programming Tools (p25-26) | | |
| | 8/2 | Problem Solving<br>o VB Problem Solving<br>o Algorithms<br>o Pseudocode | | | |
| 2 | 9/3 | Labor Day | | | |
| | 9/5 | II. VB Programming Basics | 2.1 Visual Basic Objects (p36-47) | Practice Problems p.1: | |
| | | • Objects & Events<br>o Objects<br>o Events Procedures | 2.2 VB Events (p50-58) | 1,2 | |
| | 9/7 | • Number Type<br>o Arithmetic Operations<br>o Variables<br>o Relational Operators<br>o Built-in Functions | 2.3 Numbers (p63-69) | Exercises 2.1:<br>1,3,5<br>30, 31, 37<br>Exercises 2.2:<br>1,3,5,7<br>37, 39, 41 | PA 1 (2.1), |
| 3 | | I. VB Programming Basics | 2:4 rings p 77-84) | rac ice ro ems 2.3: | |
| | | Object & Events<br>o Objects<br>o Event Procedures | | Exercises 2.3: | |
| | | Number Type<br>o Arithmetic Ops<br>o Variables<br>o Built-in Functions | | 1,3,5<br>23,25,27<br>35,39 | PA 2 (2.3) |
| | | String Type<br>o Literals & Variables<br>o Concatenation<br>o Declaring Strings<br>o Scope of Variables<br>o Text Box for I/0<br>o *ANSI (ASCII)* Character Set<br>o String Relationships<br>o String Functions | ring e a ions ips (p84) | Practice. Probems 2.4: | |
| | 9/19 | | String Functions (p85-89) | 1,2,3<br>Exercises 2.4: | |

| | | | | | |
|---|---|---|---|---|---|
| | | | | 1,3,5,7<br>17,19,21,23<br>53,55 | |
| | 9/21 | Test 1 | | | PA 3 (2.3) Example |
| | | Review Questions<br>Practice Problems 2.3<br>Programming assignments 1, 2, 3 | | | |
| | | Answers | | | |
| 5 | 9/24 | • Input and Output<br>Reading from File<br>App.Path method<br>Input from Input Box<br>Continuation Character<br>Output to Printer | 2.5 Input & Output (p96-103)<br>Message Box (p103)<br>Continuation Char p104)<br>Output to Printer (p104-107) | Practice. Problems. 2.5:<br>• 1,2<br>Exercises 2.5:<br>1,3,5,7<br>15,17,19,21<br>29,31 | |
| | 9/26 | | | | |
| | 9/28 | | | | PA 4 (2.4) |
| 6 | 10/1 | General Procedures<br>Sub Procedures<br>Procedures with Arguments | genera Procedures (p115-120)<br>ariables & Expressions As Arguments (p 120-122)<br>Passing values Back From Sub Procedures (p122-125) | Practice. ro ems.<br>• 1<br>Exercises 2.6:<br>1,3,5,7<br>27,29 | |
| | 10/3 | | | | |
| | 10/5 | | | | PA5 (2.5) |
| 7 | 10/8 | Discoverer's ay o C asses | | | |
| | 10/10 | . Passing Values Back<br>Function Procedures<br>o Built-in Functions<br>o C to F Conversion<br>o lb to kg Conversion<br>o Hypotenuse of Rt Trianglel | Function Procedures (p125-130) | Practice Problems.<br>2<br>Exercises 2.6:<br>33,35,3 | |
| | 10/12 | Test 2 (Strings and General Procedures) | | | |
| | | Review Questions<br>More on Procedures | | | |
| 8 | 10/15 | III. Cont olling rogram Flow<br>Decision Structure<br>o If-Else Structure<br>o Logical Operators<br>o Compound Conditions<br>o If-Else-If Structure | 3.1 Decision Structures (p146-148)<br>Logical Operators (p152-155) | France. Problems. 3.1:<br>• 1, 3<br>Exercises 3.1:<br>1,3,5<br>11,13,15<br>21,23<br>29,31,33 | |
| | 10/17 | | | | PA6 |

| | | | | | |
|---|---|---|---|---|---|
| 9 | 110/221 | Do While Loop<br>Algorithms with Do While Loops | 3.2 Do Loop (p164-166) | Exercises 3.2<br>1,3,5<br>7,9<br>17,19,21 | |
| 10 | TOM' e urning to File | | nput & Output p96-100) | | |
| | 11/2 | it 3 Review uestions<br>more Review stions | | | (3.1) |
| | | For...Next Loop | For...Next Coop (p177-183) | Practice. Problems. 3.3:<br>• 1,2<br>Exercises 3.3:<br>1,3,5,7<br>• 15<br>17,19,21 | Extras |
| | | IV. Arrays<br>Declaring Arrays<br>Form-level Declaration<br>Form-Load Event | 4.1 Creating & Accessing Arrays (p200-209) | | |
| | 11/7 | | | Practice Problems 4.1:<br>1,2 | |
| | 11/9 | eteran's Day (No Classes) | | | |
| 12 | 11/12<br>11/14 | • Parallel Arrays<br>Array Algorithms<br>Sorting | 4.2 Sorting & Searching (p217-221) | Exercise 1<br>1,3,5<br>7,9<br>19,21,23,25 | |
| | 11/16 | | | Practice Problems 4.2:<br>Exercises 4.2:<br>1,3,5,7<br>17,19,21 | PA8 (Loop) |
| | 11 | aeching | Searching (p222-225) | | |
| | 11/21 | e<br>ForLoop ForLoo Ans)<br>DoWhileLoop DoWhileLoop (Ans) | | | |
| 14 | 1f11 | • Arrays & Sequential Files<br>. Sentinel Value | 4.3 Arrays & Sequential Files (p231-237) | Practice Provlems 4.3:<br>Exercises 4.3:<br>1,3<br>5<br>7,9,11,13<br>17 | PA9 |
| | 11/28<br>11/30 | | | | |

| | | | | |
|---|---|---|---|---|
| 15 | 12/3<br>12/5<br>12/7 | Problems with Sort and Search<br>Algorithms<br>Review & Catching Up | PA10 | |
| | | | Extras | |
| | | | | #3 |
| | | | | #4 |
| 16 | 12/10 (mon) | **Final Exam:** | | Extra |

R.K. Matsuyama, Jr.

Chaminade University of Honolulu
Computer Information Systems
August 2001

CIS 150 Questionnaire

Date Last 4-digits of your SSN

Age Sex Ethnicity

Major Minor

When did you start at Chaminade?

When do you expect to graduate?

*The following questions refer to some of the topics that will be covered in this course. The questions are designed to assess how successfully the course will achieve its intended objectives. Please answer the questions to the best of your ability. If you do not have any idea of what a question is asking, indicate accordingly.*

1. (Hardware) What are the essential hardware components of a computer?

2. (Software) Name and explain the various types of software used by the computer?

3. (Algorithm) What is the difference between a computer program and an algorithm? Illustrate with an example.

4. (Modular programming) Explain, using an example, what is meant by the *modular* approach to program development.

5. (Array average) Given a set of integers in an array, write a short algorithm which finds the average of those numbers.

6. (Maximum) Given a set of integers in an array, write a short algorithm which finds the maximum value in the set.

7. (Event-driven program) What is an event-driven program? Explain with an example.

8. (VB Program 1) Write a short program in Visual Basic which finds the average of all consecutive integers from 1 to 100, inclusive.

9. (Problem 1) Solve the following problem. For each rented car, a rental company charges a basic fee of $10 per day. In addition, it charges 10 cents per mile for the first 50 miles, and 5 cents per mile for any mileage beyond 50 miles. Suppose you travel a distance of 450 miles in two days in your rented car. What is the total bill for the rental?

10. (Problem 2) The local Kinko has a promotional sale for copying, in which first 100 pages are charged at the rate of 8 cents a page, and anything beyond that is charged at the rate of 5 cents per page. The sales tax is 5% of the total printing cost. If you have 300 pages to be printed, what will be your total bill, including the tax?