

CALIFORNIA STATE POLYTECHNIC UNIVERSITY, POMONA

ETE 412

COURSE OUTLINE

Course Information	ABET Unit Classification (4 Quarter Units)
Department: Engineering Technology Course Number: ETE 412/412L Course Title: Introduction to Windows Programming/Lab Revision Date: 12/14/04 Revised by: Lyle B. McCurdy Compliant: Catalog 2004/05	Math: Basic Science: Engineering Topics: 4 <i>Contains significant design content:</i> No Other: Curriculum Designation: Elective

I. Catalog Description

ETE 412/412L Introduction to Windows Programming/Lab (3/1)

Introduction to Windows application programming using API functions -- menus, controls; use of class libraries. 3 lectures/problem-solving and 1 three-hour laboratory. Prerequisites: ETT 312 or equivalent.

II. Prerequisites and Co-requisites

Students are expected to have a working knowledge of C and C++ class-oriented programming (oop).

III. Textbook and/or Other Required Material

Deitel et. al., Visual C++.NET How To Program w/CD, Prentice Hall, 2004, ISBN 0-13-437377-4, or equivalent.

IV. Course Objectives

After completing this course the student will be able to:

1. Use MS Visual .NET to write C and C++ object-oriented programs using Windows API, MFC, and Framework Classes using multiple file programming techniques.
2. Use Windows Framework Forms and Controls to create practical Window application programs that utilize menus, push buttons, check boxes, timers, etc.
3. Write Windows programs that include the use of user-written dynamic link libraries, threads, and file I/O.
4. Write Windows programs that utilize XML (Extensible Markup Language).
5. Work in teams and to write effective software-oriented laboratory reports that meet professional writing standards.

V. Expanded Course Description

1. **Introduction to API Windows code** (2 weeks)
API menus and basic Window controls; the message loop and Win functions; program modularization.
2. **Introduction to Windows programming using MFC and Framework Classes** (3 weeks)
.NET Framework Classes; Windows forms and controls; events, timers, buttons, check boxes, etc.
3. **Framework Class and form controls, continued** (2 weeks)
Check boxes, Group boxes, Tab controls, MDI and visual inheritance.
4. **Dynamic link libraries, threads and synchronization, and file I/O** (2 weeks)
Write Windows programs using DLLs, threads, files, and streams.
5. **Advanced Windows topics** (1 week)
Write Windows programs using Extensible Markup Language procedures.

VI. Class/Laboratory Schedule

Lecture: Two 75 minute sessions per week
 Lab: One 3 hour session per week.

VII. Contribution of Course to Professional Component

Lecture: Students learn to use C and C++ techniques to write object-oriented programs using Windows API, MFC and Foundation Classes.unctions, and linked lists.

Lab: A wide range of programming techniques related to C++ programming will be studied in lab. Students also learn to properly document reports for programming applications, including use of flow charts and instructions in formal laboratory reports that meet professional writing standards.

VIII. Evaluation of Students

The instructor evaluates outcomes using the following methods: homework assignment submittals, midterm and final exams, one-on-one discussions during office hours, programming experiments, and laboratory reports.

The student grades are typically based on the following factors: quizzes, homework, midterm exam and final Exam.

IX. Relationship of Course to Program Outcomes

Crse Obj	Program Outcomes										
	(a) Use of modern tools of discipl	(b) Use of math, science, Engg & Tech	(c) Do experi- ments	(d) Dsn of sys & compo nents	(e) Work on teams	(f) Do Tech probs	(g) Eff Com	(h) Life- long learn	(i) Prof, ethics, social resps	(j) Prof, soc, globl, diversity	(k) Qual, Cont impr, timeli ness
1				X		X					
2				X		X					
3				X		X					
4	X					X					
5	X				X		X				

X. Typical Laboratory Experiments. Here, the students are expected to utilize the MS Visual .NET Compiler in console mode to write object-oriented programs that utilize the C++ language. The following labs are oriented to achieve this purpose:

- Lab 1.** Introduction to Windows API programming -- a menu-driven, multi-file “Hello” program using traditional API functions. Formal software-oriented laboratory report required.
- Lab 2.** Using MFC to write a menu-driven “Hello” program using MS Visual C++ visual techniques. Formal software-oriented laboratory report required.
- Lab 3.** Using MS Foundation Classes and .NET to write a Windows Form-oriented program with buttons, menus, check boxes, etc. Formal software laboratory report required.
- Lab 4.** Using MS Foundation Classes and .NET to write a Windows multiple-document-interface oriented program that utilizes threads and timers. Formal software laboratory report required.
- Lab 5.** Using MS Foundation Classes and .NET to write a Windows Form-oriented program using visual inheritance between different pages of the display. Formal software laboratory report required.