

**PALOMAR COLLEGE
COURSE OUTLINE OF RECORD
FOR DEGREE CREDIT COURSE**

(FORM VERSION 5/95)
(DATE REVISED 3/5/98)

TRANSFER COURSE A.A. DEGREE
(CHECK ALL THAT APPLY)

COURSE NUMBER AND TITLE: CSIS 288 - WINDOWS PROGRAMMING II

UNIT VALUE: 3

MINIMUM NUMBER OF SEMESTER HOURS: 80

BASIC SKILLS REQUIREMENTS: Appropriate Language and Computational Skills

ENTRANCE REQUIREMENTS:

Prerequisite: CSIS 280 and CSIS 285
Corequisite: None
Recommended Preparation: None

SCOPE OF COURSE: Windows programming using the WIN32 API for writing applications that use multitasking, threads, synchronization, and structured exception handling. Covers implementation of Dynamic Link Libraries (DLLs), Graphic Device Interface (GDI) optimization, and creation of Help files. Includes a detailed study of the Microsoft Foundation Class (MFC) Library. Presents techniques to add Object Linking and Embedding (OLE) functionality to Windows applications.

SPECIFIC COURSE OBJECTIVES: The successful student will:

1. Demonstrate correct use of the WIN32 API.
2. Implement multitasking, multithreading, and structured exception handling.
3. Explain how DLLs are used in Windows.
4. Implement DLLs.
5. Determine when it is appropriate to create a DLL vs a static library.
6. Optimize the GDI.
7. Calculate raster operation (ROP) codes to obtain desired drawing output.
8. Modify and display bitmaps.
9. Display device independent bitmaps (DIBs).

10. Convert between bitmaps and DIBs.
11. Build Help files.
12. Implement Help within an application using WinHelp.
13. Describe the purpose and contents of the MFC Library.
14. Describe the relationship between MFC and the Windows API and SDK.
15. Describe the CWinApp object in MFC programs for Windows.
16. Describe the CWnd class and the CDC class in MFC.
17. Describe the document/view architecture used by MFC.
18. Describe the MFC command routing scheme.
19. Describe the command targets available under the MFC command routing scheme.
20. Describe three categories of collection classes in MFC and their uses.
21. Describe the MFC concept of serialization of data.
22. Describe the Windows GDI routines and their uses.
23. Describe the different kinds of GDI objects.
24. Use Windows GDI routines to generate graphic output in an MFC application.
25. Add OLE functionality to Windows applications.
26. Use MFC-based facilities to create OLE Container and Server Applications.

CONTENT IN TERMS OF SPECIFIC BODY OF KNOWLEDGE:

- I. WIN32 API
 - A. Portability
 - B. The Registry
 - C. Structured Exception Handling
 - D. Process Management
 - E. Threads and Synchronization
 - F. Scheduling

- II. Dynamic-Link Libraries (DLLs)
 - A. DLL Overview
 - B. Building a DLL
 - C. Handling DLL Issues

- III. Optimizing the Graphics Device Interface (GDI)
 - A. GDI Overview
 - B. Implementing Drawing Operations
 1. Bitmaps
 2. Raster Operation (ROP) Codes
 3. Device-Independent Bitmaps (DIBs)

- IV. Building Help Files
 - A. Invoking WinHelp
 - B. Overview of Help Systems
 - C. Creating the Help Topics
 - D. Creating the Help Project Files
 - E. Building and Debugging the Help File

Microsoft Foundation Class (MFC) Application Framework

- V.
 - A. MFC Overview
 - B. The Structure of an MFC Application for Windows
 - 1. Definition of a Window Class
 - 2. SDK Wrappers
 - 3. Layers of API
 - 4. Introduction to the CWinApp and CWnd Classes
 - 5. Device Contexts: Introduction to the CDC Class
 - C. The Document/View Architecture
 - D. Command Routing in MFC
 - E. Data Storage
 - 1. Collection Classes
 - 2. Files and Serialization
 - 3. Storing and Retrieving Data
 - F. Graphic Output
 - 1. The Graphics Device Interface (GDI)
 - 2. Overview of the GDI Routines for Graphic Output
 - 3. Drawing Routines: Cpoint, Crect, and Csize Classes
 - 4. The Different Kinds of GDI Objects
 - 5. Working with GDI Objects Within a Device Context
 - G. AppWizard
 - H. Menus and Commands
 - I. ClassWizard
 - J. Toolbars
- VI. Object-Linking and Embedding (OLE)
 - A. OLE Overview
 - B. The Component Object Model (COM)
 - C. Structured Storage
 - D. Serialization
 - E. Uniform Transfer Model
 - F. Data Objects and the Clipboard
 - G. Drag and Drop
 - H. OLE Container/Server
 - I. Programming the OLE 2.0 User Interface

REQUIRED READING:

Porter, Anthony. C++ Programming for Windows. Berkeley: Osborne-McGraw Hill, 1993.

SUGGESTED READING:

Petzold, Charles. Programming Windows 3.1. Third Edition. Redmond: Microsoft Press, 1992.
 Conger, James. Windows API Bible. Corte Madera: Waite Group Press, 1992.
 King, Adrian. Inside Windows 95. Redmond: Microsoft Press, 1994.

REQUIRED WRITING:

Weekly programming assignments will average one-half to one page of writing per week.

OUTSIDE ASSIGNMENTS:

Students are expected to spend a minimum of three hours per unit per week in class and on outside assignments, prorated for short term classes.

Outside assignments will include completion of lab work, assigned readings, and programming assignments.

INSTRUCTIONAL METHODOLOGY: Check the following that apply:

- lecture
- laboratory
- lecture/laboratory combination
- directed study

This course may be offered as a distance education course and meets Title 5 regulations 55370, 55372, 55374, 55376 and 55378. Yes__ No x

If yes, check all that apply.

- telecourse
- mediated instruction
- computer assisted instruction

GRADING POLICY AND STANDARDS:

Programming Assignments	40%
Mid-term	25%
Final	35%

Is course repeatable for reason(s) other than deficient grade? Yes__ No x

Number of times course may be taken for credit: 1 If yes, identify specific provision of Division 2 section(s) 55761-55763 and 58161 which qualifies course as repeatable.

CONTACT PERSON: Richard Stegman x 2769

SIGNATURES ON FILE
