MISSOURI WESTERN STATE UNIVERSITY
COLLEGE OF LIBERAL ARTS AND SCIENCES
DEPARTMENT OF COMPUTER SCIENCE, MATHEMATICS, AND PHYSICS

CLASS SYLLABUS

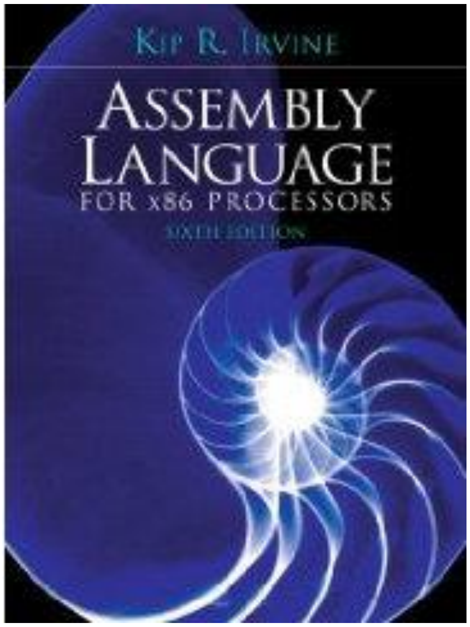
I.	Course Number	Course Name	Schedule	Credit
	CSC384	Assembly Language	MW 1:00PM-2:20PM	3

II. Prerequisites: CSC245 or CSC254 with a grade of C or better.

III. Course Description: An investigation of the logical basis of a particular computer from the programmer's viewpoint. Machine representation of numbers and characters, instruction formats, machine operations and addressing techniques will be covered.

IV. Text: Intro.TO 80x86 Assembly Language, Detmer, Edition 3rd 15, Jones+Bart,

ISBN 9781284036121



.

V. Course Objectives:

a. The student learns the structure and organization of a computer system, by learning and using the assembler language of the system.

b. The student will examine the computer architecture of the Intel micro-processors.

VI. Course Outline:

I. Basic Concepts

- A. ASM vs. High-level Languages
- B. Virtual Machine Concept (Optional)
- C. Data Representation
  - A. Binary and Hexadecimal Numbers
  - B. Character Codes
  - C. Complements of Binary and Hexadecimal Numbers
  - D. Addition and Subtraction of Complemented Numbers
  - E. Other Systems for Representing Numbers
- D. Boolean Operations

II. IA-32 Processor Architecture
- A. Basic Microcomputer Design
- B. IA-32 Processor Architecture
  - A. Pipelining and Superscalar Designs (Optional)
  - B. CISC vs. RISC Designs (Optional)
- C. IA-32 Memory Management
- D. Components of an IA-32 Microcomputer
- E. Input-Output System

III. Assembly Language Fundamentals
- A. Basic Elements of Assembly Language
  - A. Integer Constants and Expressions
  - B. Character and String Constants
  - C. Reserved Words and Identifiers
  - D. Directives and Instructions
  - E. Labels
  - F. Mnemonics and Operands
  - G. Comments
  - H. Examples
  - I. Program Template
- B. How to Assemble, Link, and Run a Program
- C. The Assembler Listing File and Map File
- D. Data Definition Directives
- E. Symbolic Constants

IV. Data Transfers, Addressing and Arithmetic
- A. Data Transfer Instructions
- B. Addition, Subtraction
- C. Data Related Operators and Directives
- D. Indirect Addressing and Pointers

V. Branching and Looping
- A. JMP Instruction
- B. LOOP Instruction
- C. Array Examples Using LOOPs

VI. Procedures
- A. Linking to an External Library
- B. Stack Operations
- C. Defining and Using Procedures
- D. Program Design Using Procedures

- VII. Conditional Processing
  - A. Boolean and Comparison Instructions
  - B. Conditional Jumps
  - C. Conditional Loop Structures
  - D. Conditional Structures
- VIII. Integer Arithmetic
  - A. Shift and Rotate Instructions
  - B. Shift and Rotate Applications
  - C. Multiplication and Division Operations
  - D. ASCII and Unpacked Decimal Arithmetic
  - E. Packed Decimal Arithmetic
- IX. Strings and Arrays
  - A. Using String Primitive Instructions
  - B. Selected String Procedures
  - C. Two Dimensional Arrays
  - D. Binary Search
- X. Macros
  - A. Defining Macros
  - B. Invoking Macros
  - C. Conditional Assembly Directives
- XI. Floating-Point Processing
  - A. Floating-Point Binary Representation
  - B. Programming with Floating-Point Instruction Set
  - C. Floating-Point Emulation
  - D. Floating-Point and In-Line Assembly

- XII. Input/output — BIOS Level Programming (Optional)
  - A. Keyboard Input
  - B. Video Programming Using INT 10h
  - C. Drawing Graphics Using INT 10h
  - D. Memory Mapped Graphics
  - E. Mouse Programming