



Ahsanullah University of Science and Technology

Department of Computer Science and Engineering

Course Outline

Course No:	CSE2214
Course Title:	Assembly Language Programming Lab
Credit Hour:	1.5
Semester(Session):	Spring 2020
Student Year & Student Semester:	2 nd Year, 2 nd Semester
Course Teacher(s):	Tahsin Aziz Samiha Sharmin Shimmi A.K.M. Amanat Ullah

Course Objective:

This course introduces the students with assembly language programming. The course is designed based on the popular Intel 8086 microprocessor and provides a good understanding of the microprocessor operation at the address, data, and control level. The course also covers the software part through the teaching of assembly language programming techniques. Students will translate a number of small C/C++ programs into assembly language and learn to trace and debug at the assembly level. The knowledge of how C/C++ constructs are translated to execute on hardware, simple hardware operations and interrupt handling are crucial building blocks for the Operating Systems and Computer Architecture courses.

This lab complements the *Computer Architecture (CSE2213)* course.

Preferred Programming Language/Tools: 8086 Assembly Language.

Text/ Reference books:

- "Assembly language Programming and Organization of the IBM PC" written by "Charles Marut and Ytha Yu". Publisher: McGraw-Hill, Inc, 1992.

Session Plan:

Week	Topics/Contents
01	<i>Microcomputer Systems</i> Components of a microcomputer system, Instruction execution, I/O devices, Programming languages.
02	<i>Representation of Numbers and Characters</i> Number systems, Conversion between number systems, Addition and subtraction, Representation of integers in the computer, Character representation.
03	<i>Organization of the IBM Personal Computers</i> Intel 8086 family of microprocessors, Organization of the 8086/8088 microprocessors, Organization of the PC.
04	<i>Introduction to IBM PC Assembly Language</i> Assembly language syntax, Program data, Variables, Named constants, Few basic instructions, Translation of high-level language to assembly language, Program structure, Input and output instructions.
05	<i>The Processor Status and the Flags Register</i> The flags register, Overflow, Instructions affecting the flags.
06	<i>Mid-Term Examination + Review of Previous Classes</i> The mid-term test will be held based on the topics covered up to 5 th week. A review class will be taken on session 1 to session 5
07	<i>Flow Control Instruction</i> An example of the jump, Conditional jumps, The JMP instruction, High-level language structures.
08	<i>Flow Control Instructions</i> An example of a loop, For loop, While loop, Until loop structures.

09	<i>Logic, Shift and Rotate Instructions</i> Logic instructions, Shift instructions, Rotate instructions, Binary, and hex I/O.
10	<i>The Stack and Introduction to Procedures</i> The stack, A stack application, Terminology of procedures, CALL and RET.
11	<i>Multiplication and Division Instructions</i> MUL and IMUL, DIV and IDIV, Sign extension of dividend, Decimal input, and output procedures.
12	<i>Arrays and Addressing Modes</i> One dimensional array, Addressing modes, Two-dimensional arrays, Based indexed addressing modes, The XLAT instruction.
13	<i>The String Instructions</i> The direction flag, Move a string, Store string, Load string, Scan string, Compare string.
14	<i>Final Term Examination + Review of the Full Course + Pending Tasks</i> A review class will be taken on session 1 to session 5 and session 7 to session 12.

Note: *This Session Plan is subject to change. Course teacher will slow down or speed up each chapter to meet the needs of students.*

Marks Distribution:

Attendance and Class Performance	20
Assignment (Home Assignment/ Offline/ Class Assignment/Online)	40
Lab Quiz (Mid Term and/or Term Final)	40
Total	100

In each lab class, some related programming problems will be given to the students that should be solved in the lab. Again, some problems will be given on the same topic as individual assignments that will be checked in the next lab class.