ALLAMA IQBAL OPEN UNIVERSITY, ISLAMABAD
(Department of Computer Science)

WARNING

1. **PLAGIARISM OR HIRING OF GHOST WRITER(S) FOR SOLVING THE ASSIGNMENT(S) WILL DEBAR THE STUDENT FROM AWARD OF DEGREE/CERTIFICATE, IF FOUND AT ANY STAGE.**
2. **SUBMITTING ASSIGNMENTS BORROWED OR STOLEN FROM OTHER(S) AS ONE'S OWN WILL BE PENALIZED AS DEFINED IN "AIOU PLAGIARISM POLICY".**

Course: Computer Organization & Assembly (3453)

Level: BS (CS) **Semester: Spring, 2014**

Total Marks: 100 **Pass Marks: 50**

ASSIGNMENT No. 1

(Unit 1–4)

Note: All questions are compulsory. Each question carries equal marks.

Q.1 a) In which kind of problem domains, assembly language can be readily applied? Give arguments to support your answer.

b) Describe the major differences between machine and Assembly Language instructions? Give two examples.

Q.2 a) Explain the working of Segment resister of the CPU?

b) Convert the following segment offset addressed to absolute addresses:
0800:0100 h
07F2:101E h
0040:0049 h
03ED:FFFF h

Q.3 a) What is the role of FLAG register during the execution of program?

b) Describe the typical arrangement of elements found in IBM PC in first 1 Mega Byte?

Q.4 a) Define any array of 100 elements of type and initialize with the maximum value an element can store.

b) Initialize two string arrays with your name and father's name respectively.

c) Consider the double word 12347856H? How this double word will be stored in Memory. What data type should be selected to store this value?

Q.5 a) Write down the basic structure of an assembly program that include data and code segments.

b) How a program is loaded into memory and then executed? Write all steps in sequence.

ASSIGNMENT No. 2

(Unit 5–8)

Total Marks: 100 **Pass Marks: 50**

Q.1 a) Write instructions that jump to a label named Target of bits 0, 1 and 2 in the AL register are all set (the remaining bits are unimportant).

b) Write instructions that will jump to a label named Target if either bit 0, 1 or 2 is set in the AL register (the remaining bits are unimportant).

c) Clear bits 4-6 in the BL register without affecting any other bits.

d) Set bits 3-4 in the CL register without affecting any other bit.

Q.2 Explain all the addressing modes of 80x86 based microprocessor that deal with memory. Memory may be referenced using named or un-named methods. Give proper examples to explain each.

Q.3 a) Write a program that reads a series of ten lowercase letters from input (without displaying it), converts each character to uppercase and then displays the converted character.

b) Write a program that displays a string using INT 21h function 9.

Q.4 a) Write a program that inputs a string using DOS function 0Ah. Limit the input to ten characters. Redisplay the string backwards.

b) Write a program that inputs a string of up to 80 characters using DOS function 3Fh. After the input, display a count on the screen of the actual number of characters typed by the user.

Q.5 a) Write a program that inputs the month, day and year from the user. Use the values to set the system date with DOS function 2Bh. Hint: Use the Readint function from the book's link library to input the integer values. (Under Windows NT/200, you must have administrator privileges to run this program).

b) Write a program that uses DOS function 2Ah to get and display the system date. Use the following display format: yyyy-m-d.

3453 Computer Organization and Assembly

Recommended Book: Assembly Language for Intel-Based Computers, by **Kip R. Irvine**

Course Outlines:

Unit # 1 Introduction

a. Context of Assembly Language
   Introduction to assembly language, Assembly Language Applications
b. Data representation
   Binary numbers, Converting Binary to Decimal, Hexadecimal Number:-. Signed Numbers, Character Storage,
c. Introducing Assembly Language
   Language Instructions, A Sample Debug Program, Debug Commands

Unit # 2. Hardware and Software Architecture

a. 16-BIT Intel Processor Architecture
   Central Processing Unit, Registers, Status and Control Registers, Flags; Instruction Execution Cycle, Intel Microprocessor Family,
b. 32-BIT Intel Processor Architecture
   Improved Execution Cycle, 32-bit Register Set
c. Operating System and Memory
   Memory Architecture, Video display, Absolute Address Calculation-Video Adapter, Memory (RAM), Video RAM,

Unit #3. Assembly Language Fundamentals

a. Basic Elements
   Constants and Expressions, Statements, Names, Sample Hello Program
b. Assembling Linking and Debugging
   Microsoft Assembler (MASM)
c. Data Allocation Directive
   Define Byte (DB), Define Word (DW), Define Doubleword (DD),
d. Symbolic Constants
   Equal-Sign Directive, EQU Directive, TEXTEQU Directive

Unit # 4. Instructions

a. MOV Instruction
   Operands with Displacements, XCHG Instruction
b. Arithmetic Instructions
   INC and DEC Instructions, ADD Instruction, SUB Instruction, 1 lags Affected by ADD and SUB
c. Basic Operand Types
   Register Operands, Immediate Operands, Direct Operands, Direct Offset Operands,
d. Extended Addition and Subtraction
   ADC Instruction, SBB Instruction
e. Multiplication and Division
   MUL Instruction, IMUL Instruction, DIV Instruction, IDIV Instruction, CBW, CWD, CDQ, and CWDE Instructions, Preventing Divide Overflow

Unit # 5 Using the Assembler.

a. More About the Assembler and Linker ;
   Source Listing File, Map File, Assembling and Linking with MS-DOS Batch Files, Memory Models, Target Processor Directives
b. Operators and Expressions
   Arithmetic Operators, OFFSET, SEG, PTR, LABEL, AND EVEN
c. IMP and LOOP Instructions
   JMP Instructions, LOOP Instruction LOOP, LOOPW, LOOPD Instructions
d. Indirect Addressing
   Indirect Operands, Based and Indexed Operands, Base-Index wiUi-Displacement

Unit # 6. Conditional Processing

a. Boolean and Comparison Instructions
   The F'ags Register, AND Instruction, OR Instruction, XOR Instruction," NOT Instruction, NEG Instruction, TEST Instruction, CMP Instruction, CMPXCHG Instruction
b. Integer Arithmetic
   Shift and Rotate Instructions, SHL Instruction, SHLD/SHRD Instructions, SHR Instruction, SAL and SAR Instructions, ROL Instruction, ROR Instruction, RCL and RCR Instructions,
c. Sample Applications
   Shifting Multiple Bytes on the 8086, Fast Multiplication and Division

Unit # 7. **Conditional Jumps**

a. Code Generation for Conditional Jumps, Conditional Jump Examples
b. Conditional Loops
   LOOPZ and LOOPE Instructions, LOOPNZand LOOPNE Instructions
c. High-Level Logic Structures
   Simple IF Statement, Compound IF Statement, WHILE Structure-, REPEAT-UNTIL Structure, CASE Structure, Table of Procedure Offsets.

Unit # 8. Software Interrupts

a. INT Instruction, Redirecting Input-Output
b. MS-Dos Function Calls
   Output Functions, Input Functions, Date/Time Functions,
c. BIOS-Level Keyboard Input (INT16H)
d. BIOS-Level Video Control (INT 10H)
e. Disk Storage Fundamentals
f. Drive and Directory Manipulation

Unit # 9. Procedures and Interrupts,

a. Stack Operations
b. Procedures,
c. Procedure Parameters
d. Structures
e. Macros
   Macros with Parameters, Defining a Macro, Example: mDisplayiStr, Macro, mGotoRowCol macro