Program: BC (CS)
Subject: Microprocessor & Assembly Language
Major Assignment Mid-Term
Course Code: CSC-304
EDP Code: 102007054
Semester: Summer 2020

Q.1 Solve each of the following:

a. $64_{10} = (\ ?\ )_2$
b. $01111111_2 = (\ ?\ )_{10}$
c. $4D7F_{16} = (\ ?\ )_{10}$
d. $128_{10} = (\ ?\ )_{16}$
e. $3A6F_{16} = (\ ?\ )_2$
f. $1100001111100101_2 = (\ ?\ )_{16}$
g. $11111111_2 = \pm (\ ?\ )_{10}$ hint: [use 2's complement form]
h. $-16_{10} = (\ ?\ )_2$ hint: [use 2's complement form]
i. $01111111_2 - 00000111_2$ hint: [use 2's complement form]
j. $6D_{16} - 3F_{16}$ hint: [use 2's complement form]

Q.2 Write short note on each of the following:

a. Embedded systems
b. Device driver
c. Virtual machine concept
d. Instruction execution cycle
e. Motherboard Chipset
f. Access levels for input–output operations
g. Basic parts of an assembly language instruction

Q.3 Differentiate between each of the following:

a. Assembly language and high-level language

b. Protected mode and real address mode

c. Assembler and linker

d. Instruction and directive

e. Code label and data label

f. Line comment and block comment

g. Equal-sign directive and EQU directive

Q.4 Give answer to each of the following

a. Explain the concept of portability as it applies to programming languages.

b. Why would a high-level language not be an ideal tool for writing a program that directly accesses a particular brand of printer?

c. Why was Unicode invented?

d. If W = 11101100, X = 00010011, and Y = 00111100, then find Z = W $\vee$ X $\wedge$ $\neg$Y.

e. Create a truth table to show all possible inputs and outputs for the Boolean function described by $\neg$(A $\vee$ B)

f. Why does memory access take more machine cycles than register access?

g. Discuss the basic program execution registers used in x86 32-Bit processors.

Q.5 Discuss the following MASM directives in detail:

INCLUDE	.386	.MODEL	.STACK	PROTO
.DATA	.CODE	PROC	ENDP	END

Q.6 a. Write a program that calculates the following expression: *A = (A + B) − (C + D)*

b. Show the order of individual bytes in memory for the following doubleword variable using little endian order: *dval DWORD 12345678h*

c. Write a statement that causes the assembler to calculate the number of bytes in the following string, and assign the value to a symbolic constant named StringSize:

string1 byte “Assembly language is easy”, 0

d. Write a program that performs arithmetic operations on different register operands and stores the result in memory. Give stepwise explanation of each statement.