SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY

SYSTEMS PROGRAMMING CSC 2209

TEST 1

Answer all questions
Time allowed: 1 Hour

1. (i) What is Software?

Instructions and data that the computer manipulates to perform various data processing tasks.

(ii) How does "Systems Software" differ from Development Software?

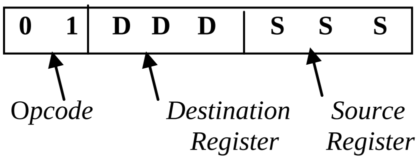
The system software are programs whose purpose is to make more effective use of the computer. They control the operation of the machine and carry out the most basic functions the computer performs. They control the way in which the computer receives input, produces output, manages and stores data, carries out or executes instructions of other programs etc. They support the operation and use of the computer itself rather than any particular application.
Development Software is used to create, update and maintain other programs e.g. programming languages.

(iii) Give examples of three Systems Software and explain their uses.

- ***Compilers: Translate High Level Languages to Machine Language.***
- ***Assemblers: Translate Assembler Language to Machine Language***
- ***Loaders: Load programs in memory for Execution***
- ***Linkers: Combine different programs and prepare them for execution.***
- ***Macro processors: Allow programmers to use abbreviations***
- ***Operating systems and file systems: Allow flexible storing and retrieval of information.***

2. Assume the following Intel 8085 Instruction Formats:

Register to register transfer



Load accumulator from memory

0	0	1	1	1	0	1	0
Low order Address							
High Order Address							

Add immediate Data to a register

1	1	0	0	0	1	1	0
Data							

Store accumulator contents to memory

0	0	1	1	0	0	1	0
Low order Address							
High Order Address							

Write a machine language program (consecutive instructions) for the INTEL 8085 machine that picks a number from a memory location AB56H, adds a number six(6) to the picked number and stores a copy of the answer in register B and another copy in memory at a location 4567H.

	0 0 1 1 1 0 1 0	**3A**
Load the accumulator	**0 1 0 1 0 1 1 0**	**56**
with a number in memory	**1 0 1 0 1 0 1 1**	**AB**
Add 6 to the picked	**1 1 0 0 0 1 1 0**	**C6**
number	**0 0 0 0 0 1 1 0**	**06**
Store the answer	**0 1 0 0 0 1 1 1**	**47**
In register B		
	0 0 1 1 0 0 1 0	**32**
Store another copy	**0 1 1 0 0 1 1 1**	**67**
in memory.	**0 1 0 0 0 1 0 1**	**45**

3. (a) How does Assembler Language differ from Machine Language?

Assembler Language uses abbreviations for its instructions while Machine Language uses 0's and 1's for the same instructions.

(b) Explain the following:

(i) A Mnemonic

An abbreviation in Assembler Language that specifies what the instruction does.

(ii) A Directive

A Statement in Assembler Language that gives directions to the assembler during the assembly process but is not translated into machine code.

(c) Assume the following statements in C Language: int x; and x = 3;

(i) What do they mean?

Int x: reserve a memory location to store an integer.
X = 3: put an integer 3 in X (the reserved memory location)

(ii) Write corresponding Assembler Language statements for the Intel 8085 microprocessor.

X: DS 1
X: DB 3

(iii) Write corresponding Assembler Language statements for the SIC/XE microprocessor.

X: RESW 1
X: WORD 3

4. (a) How would you represent numbers +20 and -20 in the SIC/XE Format?

+20 = 10100_2 = 0000 0000 0000 0000 0001 0100 = 000014_{16}
-20 = 0000 0000 0000 0000 0001 0100
Invert bits and add 1 = 1111 1111 1111 1110 1011 + 1
1111 1111 1111 1110 1100 = $FFFEC_{16}$

(b) Write an Assembler Language Program for the SIC/XE machine that will add six numbers in an array MYARR and will store the answer at a memory location WASBIG if the sum was greater than 50 or at a memory location WASSMALL if the sum was less than 50 or at a memory location WASEQUAL if the sum was 50.

```
            LDS         #3
            LDT         #18
            LDX         #0
            LDA         MYARR, X
LOOP:       ADDR        S, X
            ADD         MYARR, X
            COMPR       X, T
            JLT         LOOP
            LDT         #50
            COMPR       A,T
            JGT         BIG
            JLT         SMALL
            STA         WASEQUAL
            J           OUT
BIG:        STA         WASBIG
            J           OUT
SMALL:      STA         WASSMALL
OUT:


MYARR:              RESW        6
WASBIG:             RESW        1
WASSMALL            RESW        1
WASEQUAL:           RESW        1
```