Q No.1	**[CLO1] (50)**

I. For each of the following statements, indicate whether it is true or false. To get credit, you must give reasons (very briefly) for your answer. (30)

   i.   In microprocessors, first in first out storage mechanism is used in stack operations.

   ii.   The size of each segment in 8086 is 16Kb.

   iii.   Base relative-plus-index addressing is useful for addressing a two-dimensional memory array.

   iv.   There are six segment registers present in 8086 microprocessors.

   v.   Indirect addressing mode is not possible in 8085.

   vi.   Sixteen address lines required to address a memory of size 32 K.

   vii. The BP or EBP register addresses data in the stack segment.

viii.   The Overflow flag is based on signed arithmetic

ix.   In 8086 for string manipulation instructions, carry flag is used.

x.   The paging unit is controlled by the contents of the microprocessor's general-purpose registers.

xi.   8-bit registers are AH, BH, CH, DH, and SI.

xii.   The segment registers are CS, DS, ES, SS, FS, and GS.

xiii.   Some real mode memory addresses are a combination of a segment address plus linear address.

xiv.   Immediate addressing manipulates constant data in a program.

xv.   Direct addressing requires 4 bytes of memory, whereas displacement addressing requires 3bytes.

xvi.   The only addressing modes not allowed with DEC are immediate or register addressing.

xvii.   The remainder after a signed division always assumes the sign of the divisor.

xviii. A 16-bit-wide memory device is often called a byte-wide memory.

xix.   8086 is a 16-bit microprocessor with a 16-bit data bus; 8088 has an 8-bit data bus.

xx.   The 16-bit multiplication always multiplies register AX by an operand with the product found in DX-AX.

xxi.   In the 32-bit division, the EDX-EAX register is divided by the operand, after which the EDX register contains the quotient and the EAX register contains the remainder.

xxii.   Opcodes that transfer data between the stack and the flags are PUSHF and POPF.

xxiii. If CS = 020AH, SS = 0801H, SI = 0100H and IP = 1BCDH the address of the next instruction is 03C6D.

xxiv. The ADD instruction allows all addressing mode.

xxv. Each statement in an assembly language program consists of four parts or fields. The rightmost field is called the label.

xxvi. The PUSHA and POPA instructions are valid in the 64-bit mode.

II. Explain the following: (16)
i. Different Busses in Microprocessor

ii. TPA

iii. Overflow Flag

iv.   Protected Addressing mode
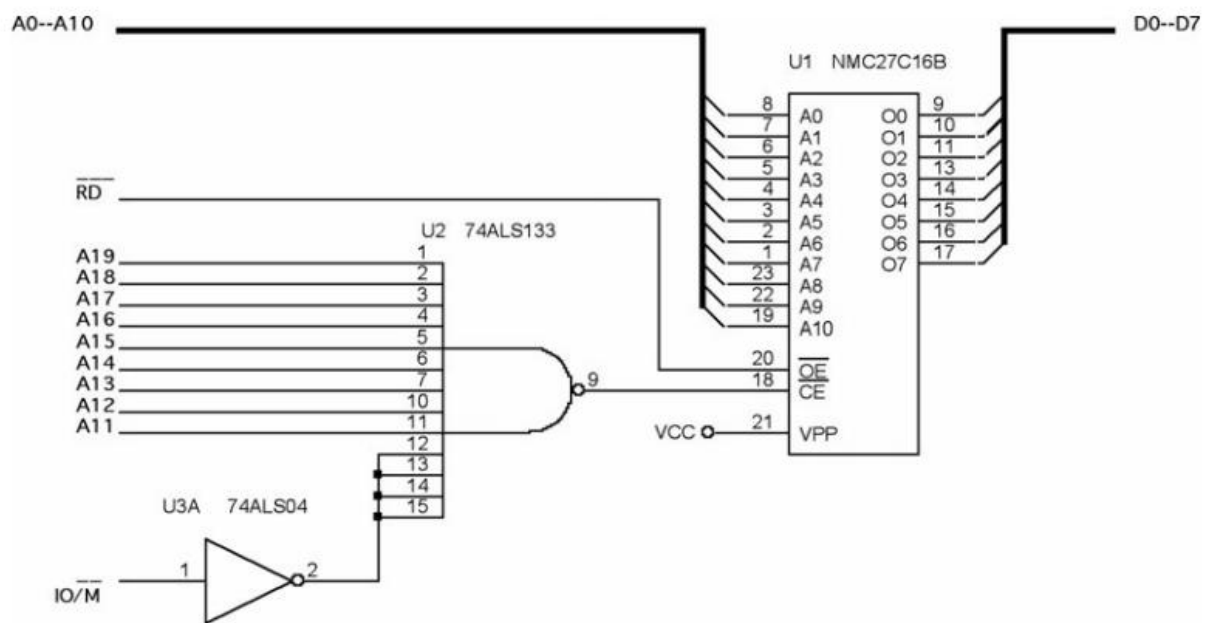
v.   Fixed and Variable addresses for I/O

vi.   Handshaking

vii.   Linear address

viii.   FPU

III.   Modify the NAND gate decoder (shown below) to select the memory for address range 40000H-407FFH. (5)

Q No.2	[CLO2] (25)

I.   **Answer the Following:** (13)

i.   Suppose that DS = 0200H, BX = 0300H, and DI = 400H. Determine the memory address accessed by MOV AL,[1234H] instruction, assuming real mode operation. (2**)**

ii.   Suppose that EAX = 00001000H, EBX = 00002000H, and DS = 0010H. Determine the addresses accessed by the following instructions, assuming real mode operation: (4**)**
(a) MOV [EAX+2*EBX],CL
(b) MOV DH,[EBX+4*EAX+1000H]

iii.   Convert an 8B9E004CH from machine language to assembly language. (2**)**
MOV BX,[BP+4C00H]

iv.   If EDX: 7FFFFFFF. What will be the values of the EDX, Carry, Overflow and Sign flags after the following instruction has executed? (2**)**
inc edx

II.   Explain the following commands: (3)

i.   MOV BX,AX

ii.   INC [BL]

iii.   SBB [DI–4], DX

III.   Correct the following code: (4)

```
INCLUDE irvine32.inc

.data
arrayA word 4Ah, 3Ch, 2h, 5 DUP (1h)
var1 word ffffh

.code
main PROC
mov cx,4Ch
exchg cx, arrayA+4
mov ax, arrayA+5
mov bl, var1
add ax, bx
call dumpregs
exit
main ENDP
```

Q No.3	[CLO2] (10)

I. **Write a sequence of instructions that cube the 8-bit number found in DL. Load DL with a 5 initially, and make sure that your result is a 16-bit number.**

II. **Write a procedure named _Str_add_ that add _n_ characters to a string.**

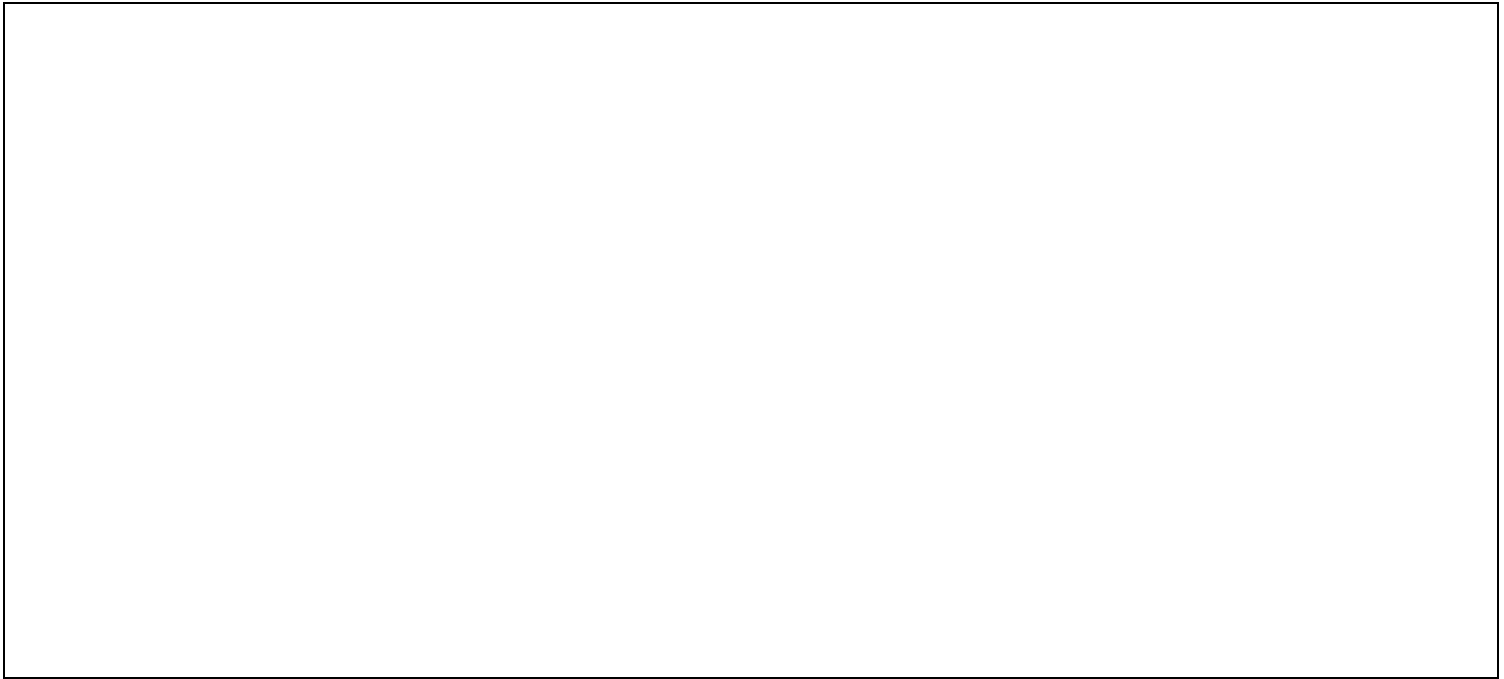| Q No.4	[CLO2] (8)

Create a program that functions as a simple Boolean calculator for 32-bit integers. It should display a menu that asks the user to make a selection from the following list:

a. x AND y
b. x OR y
c. NOT x
d. x XOR y
e. Exit program

When the user makes a choice, call a procedure that displays the name of the operation about to be performed. Continue the solution program by implementing the following procedures:

- AND_op: Prompt the user for two hexadecimal integers. AND them together and display the result in hexadecimal.
- OR_op: Prompt the user for two hexadecimal integers. OR them together and display the result in hexadecimal.
- NOT_op: Prompt the user for a hexadecimal integer. NOT the integer and display the result in hexadecimal.
- XOR_op: Prompt the user for two hexadecimal integers. Exclusive-OR them together and display the result in hexadecimal.

Q No.5	[CLO2] (7)

Use a loop with indirect or indexed addressing to reverse the elements of an integer array.

End of Exam