



# United International University

## Department of Computer Science and Engineering

CSE 236: Assembly Programming Laboratory

**Set B**

**Mid Term Lab Examination**

**Time: 60 Min**

Name:

ID:

---

1. Write down a code for printing your name using 8086 assembly

[10]

`.data`

`.code`

2. Identify the legal instructions and put a tick (✓) mark on the side of it -

[5]

- ☐ PUSH AX
- ☐ POP ES
- ☐ RET 2
- ☐ POP 2
- ☐ PUSH 2
- ☐ CALL 200
- ☐ CALL PROCEDURE 10, 20
- ☐ PUSHF
- ☐ PUSHA
- ☐ CALL FUNC

3. Write the function of each line of the following code. Two lines are done for you.

[10]

```
org 100h      ;
MOV AH, 00h   ; interrupts to get system time
INT 1AH       ; CX:DX now hold number of clock ticks since midnight

MOV AX, 10    ;
ADD AX, DX    ;
XOR DX, DX    ;
DIV CX        ;

add dl, '0'   ;
mov ah, 2h    ;
int 21h       ;
RET           ;
end main      ;
```

4. How many bits are there in the Address Bus of 8086 processor? Calculate the physical address specified by the **Segment:Offset** values given by 10F0:FF10

[5]

5. Write down one function for each of the following registers -

[5]

- SI:
- CX:
- DS:
- IP:
- AX:

6. Write down the output of following code -

[10]

```
org 100h
.data
    nl db 0dh, 0ah, '$'
.code
    mov ax, @data
    mov ds, ax
    mov cl, X      ; here X is the last digit of your student ID % 5
L2:
    push cx
L1:
    mov ah, 2
    mov dl, '*'
    int 21h
    loop L1
    mov ah, 9
    lea dx, nl
    int 21h
    pop cx
    loop L2
ret
OUTPUT:
```

7. Put a tick mark beside the valid names for variables/labels of 8086 assembly -

[5]

- ☐ goodname
- ☐ good\_name
- ☐ 1badname
- ☐ good.name?
- ☐ .badname?
- ☐ badname1
- ☐ \_good\_name
- ☐ 1goodname?
- ☐ good name
- ☐ good\$name

8. Modify the following code in assembly language to print the pattern -

[15]

```
1
1 2 3
1 2 3 4 5
```

You have to use the macro given in the code. You can add/edit/delete any lines given in the following code.

```
.data
x db 5
.code
mstar macro n
    local L1
    mov ah, 2
    mov dl, '*'
L1:
    int 21h
    dec n
    jnz L1
endm

mov dx, @data
mov ds, dx
```

9. The following algorithm may be used to carry out multiplication of two positive numbers M and N by repeated addition: [15]

```
initialize product to 0
REPEAT
    add M to product
    decrement N
UNTIL N = 0
```

Write a procedure to multiply AX by BX, and put the product in DX. The content of all the other registers should remain unchanged. You may ignore the possibility of overflow.

10. Write a program that prompts the user to type a hex number of four hex digits or less, and outputs it In binary on the next line. Suppose the user do not enter any incorrect digits and all the input letters are in Capital [20]

Sample example:

TYPE A HEX NUMBER (0 TO FFFF): 1ABC

IN BINARY IT IS 0001101010111100