Date: 02/10/20

LAB - 3 Program

+ Read an alphanumeric character & display its equivalent ASCII code at centre of the screen.

```
.model small
.data
msg1 db 0dh, 0ah, "enter alphanumeric character $"
res db 02 dup(0)
.code
mov ax, @data
mov ds, ax
lea dx, msg1
call disp
mov ah, 01h
int 21h
mov bl, al
mov cl, 4                ; 3A = 3341
shr al, cl               ; AL=3A   AL=03   BL=3A
cmp al, 0ah              ; bl = 3A    30     0F
jc digit                 ;            33     0A
ADD AL, 07H                                  ;41
digit: add al, 30h
       mov res, al
       and bl, 0fh
       cmp bl, 0ah
       jc digit1
       add bl, 07h
digit1: add bl, 30h
        mov res+1, bl

        mov ah, 00h
        mov al, 03h      ; TEXT MODE
        int 10h
```

```
mov ah, 02h          ; SET THE CURSOR POS
mov bh, 00h          ; PAGE NUMBER
mov dh, 0ch          ; ROW (00 IS TOP
mov dl, 28h          ; COLUMN VAL
int 10h

mov res+2, '$'
lea dx, res
call disp
mov ah, 4ch
int 21h

disp proc near
mov ah, 09h
int 21h
ret
disp endp
end
```

LAB-4 Program

Date: 02 / 10 / 20

* Reverse a given string & check whether it is palindrome or not.

```
.MODEL SMALL
DISPLAY MACRO MSG
        lea dx, msg
        mov AH, 09H
        int 21H
ENDM

.DATA
msg1 db 0DH, 0AH, "ENTER STRING: $"
msg2 db 0DH, 0AH, "REVERSE STRING: $"
msg3 db 0DH, 0AH, "I/P STRING IS PALINDROME.$"
msg4 db 0DH, 0AH, "NOT PALINDROME.$"
STRING db 80H DUP(?)
Rstring db 80H DUP(?)

.CODE

start: mov AX, @DATA
       mov DS, AX
       Display msg1
       mov SI, OFFSET string
       XOR CL, CL

AGAIN: mov AH, 01H
       INT 21H
       cmp AL, 0DH
       JE NEXT
       mov [SI], AL
       INC SI
       INC CL
```

Date: / /

```
        JMP AGAIN

NEXT :  MOV [SI] , Byte ptr '$'
        ; string input over...
          DEC SI
          MOV CH , CL
        ; REVERSE the string and store in Rstring
        MOV DI , OFFSET Rstring

BACK :  MOV AL , [SI]
        MOV [DI] , AL
        DEC SI
        INC DI
        DEC CH
        JNZ BACK
        MOV [DI] , Byte ptr '$'
        DISPLAY MSG2
        DISPLAY Rstring
        MOV SI , OFFSET String
        MOV DI , OFFSET Rstring

AG :    MOV AL , [SI]
        CMP AL , [DI]
        JNE FAIL
          INC SI
          INC DI
        DEC CX
        JZ SUCCESS
        JMP AG

FAIL :  DISPLAY MSG4
          JMP FINAL
```

RAJDHANI

SUCCESS : DISPLAY MSG3

FINAL : MOV AH, 4CH

INT 21H

END