ASCI.ASM

LAB6: Convert an character to asscii code

```
.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
shr al, cl
cmp al ,0ah
jc digit
ADD AL, 07H

digit: add al, 30h
       mov res, al
       and bl , 0fh
       cmp bl , 0 ah
       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  rest +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) Check wheather given String is Palindrom or Not:

```
• Mode 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: $"
  msg3 db 0DH, 0AH, " I/P String is Palindrom.$"
  msg4 db 0DH, 0AH, " Not Palindrome.$"
  String db 80H DUp(?)
  RString db 80H DUp(?)

• Code

Start: Mov AX, @Data
        mov DS, AX
        Display msg1
        ; Take String character by character

        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
          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

Success :  Display  MSg3
Final  :   MOV AH, 4CH
           INT  21H

        END
```