

27/10/20

Lab 6



Page No: 17

Date: / /

Q → WAP to display ^{ascii value of} an alphanumeric character.

• model small

• data

msg1 db 0dh, 0ah, "Enter alphanumeric char &"

res db 02 dup(0)

• code

mov ax, @data

mov ds, ax

lea dx, msg1

call disp

mov ah, 01h

int 21h

mov ~~ax~~ bl, al

mov cl, 4

shr al, cl

cmp al, 0ah

jc digit

add al, 07h

digit: add al, 30h

mov rsi, 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
int 10h
```

```
mov ah, 02h
mov bh, 00h
mov dh, 0ach
mov dl, 28h
int 10h
mov res+2, 'g'
lea dx, res
call disp
mov ah, 4ch
int 21h
```

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

Q.7 WAP to check whether string 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, "Input is palindrome string $"  
msg4 db 0dh, 0ah, "Input is 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 al, al
```

```
again: mov ah, 01h
```

```
        int 21h
```

```
        cmp al, 0dh
```

```
        je next
```

```
        mov [si], al
```

```
        inc si
```

```
        incl cl
```

```

    jmp again
next: mov [si], byte ptr '$'
    dec si
    mov ch, cl
    mov di, offset rstring
back: mov al, [si]
    mov [di], al
    dec si
    inc di
    dec ch
    jnz back
    mov [di], byte ptr '$'
    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 ch
    jz success
    jmp ag
fail: display msg4
  
```



Page No : 21

Date : . . .

jmp final

success : display msg 3

final : mov ah, 9ch

int 21h

end