**INT 10h – Video Hizmetleri (BIOS)**

**Ekranla ilgili işlemler:**

| **AH** | **Açıklama** |
| --- | --- |
| **00h** | **Video modu ayarla** |
| **01h** | **Cursor (imleç) şeklini ayarla** |
| **02h** | **Cursor pozisyonunu ayarla** |
| **03h** | **Cursor pozisyonunu oku** |
| **06h** | **Ekranı yukarı kaydır** |
| **07h** | **Ekranı aşağı kaydır** |
| **09h** | **Karakter ve özellik yaz** |
| **0Ah** | **Karakter yaz (cursor hareketi yok)** |
| **0Eh** | **TTY karakter çıkışı (BIOS Print)** |
| **13h** | **Satır yaz (birden çok karakter)** |

**INT 21h – DOS Fonksiyonları**

**DOS altında temel işletim sistemi işlemleri:**

| **AH** | **Açıklama** |
| --- | --- |
| **01h** | **Klavyeden karakter oku (echo'lu)** |
| **02h** | **Ekrana karakter yaz** |
| **09h** | **'$' ile biten string yaz** |
| **0Ah** | **Klavye ile satır oku (buffer'a yazar)** |
| **0Ch** | **Giriş işlemi (klavye buffer temizle)** |
| **4Ch** | **Programı sonlandır (DOS'a dön)** |
| **3Dh** | **Dosya aç** |
| **3Eh** | **Dosya kapat** |
| **3Ch** | **Dosya oluştur** |
| **3Fh** | **Dosyadan oku** |
| **40h** | **Dosyaya yaz** |

**INT 16h – Klavye Hizmetleri (BIOS)**

**Klavye ile ilgili BIOS düzeyi işlemler:**

| **AH** | **Açıklama** |
| --- | --- |
| **00h** | **Tuşa basılmasını bekle ve ASCII + ScanCode al** |
| **01h** | **Tuş var mı kontrol et (varsa bayrak set)** |
| **02h** | **Scan code oku (son tuşun)** |
| **10h** | **Genişletilmiş klavye servisi** |

**INT 1Ah – Zaman ve Tarih Hizmetleri (BIOS)**

| **AH** | **Açıklama** |
| --- | --- |
| **00h** | **Sistem zamanını oku** |
| **01h** | **Sistem zamanını ayarla** |
| **04h** | **Tarihi oku** |
| **05h** | **Tarihi ayarla** |

**INT 13h – Disk Hizmetleri (BIOS)**

| **AH** | **Açıklama** |
| --- | --- |
| **00h** | **Sürücü reset** |
| **02h** | **Sektör oku** |
| **03h** | **Sektöre yaz** |
| **08h** | **Disk parametrelerini oku** |

**Örnek: Klavyeden İki Rakam Al, Topla ve Sonucu Yazdır**

.model small

.stack 100h

.data

msg1 db '1. rakami girin: $'

msg2 db 13,10,'2. rakami girin: $' ; 13 ve 10 yeni satır içindir

msg3 db 13,10,'Toplam: $'

.code

main:

mov ax, @data

mov ds, ax

; 1. mesajı yaz

mov ah, 09h

lea dx, msg1

int 21h

; 1. karakteri al

mov ah, 01h

int 21h

sub al, '0' ; ASCII'den sayıya çevir

mov bl, al ; 1. rakam BL’ye

; 2. mesajı yaz

mov ah, 09h

lea dx, msg2

int 21h

; 2. karakteri al

mov ah, 01h

int 21h

sub al, '0' ; ASCII'den sayı

add bl, al ; toplam BL’de

; Toplamı ekrana yazmak için sayıdan tekrar ASCII’ye çevir

add bl, '0'

; Sonuç mesajını yaz

mov ah, 09h

lea dx, msg3

int 21h

; Toplamı yaz

mov dl, bl

mov ah, 02h

int 21h

; Programı sonlandır

mov ah, 4Ch

int 21h

end main

**Açıklama**

* int 21h ile DOS servisleri kullanılır:
  + ah = 01h: Klavyeden karakter oku (sonuç AL'de).
  + ah = 02h: Tek karakter ekrana yaz.
  + ah = 09h: $ ile biten mesajı yazdır.
* '0' karakterinin ASCII değeri 48 olduğu için sayıya çevirmek için sub al, '0' yaparız.

**Örnek Program: Gizli Karakter Girişi**

Bu program, klavyeden bir karakter alır ama ekrana yazmaz. Ardından aynı karakteri tekrar ekrana yazdırır.

.model small

.stack 100h

.data

msg1 db 'Bir karakter girin (gizli): $'

msg2 db 13,10, 'Girdiginiz karakter: $'

.code

main:

mov ax, @data

mov ds, ax

; Mesaj 1: Kullanıcıdan gizli giriş iste

mov ah, 09h

lea dx, msg1

int 21h

; AH = 08h ile karakteri oku (echo yok)

mov ah, 08h

int 21h

mov bl, al ; AL'deki karakteri BL'ye kaydet

; Yeni satıra geç

mov ah, 09h

lea dx, msg2

int 21h

; BL'deki karakteri ekrana yaz

mov dl, bl

mov ah, 02h

int 21h

; Programı sonlandır

mov ah, 4Ch

int 21h

end main

Program: Şifre Girişi (Echo = *) ve Gizli Saklama*

.model small

.stack 100h

.data

msg1 db 'Sifrenizi girin (5 karakter): $'

msg2 db 13,10,'Girilen sifre: $'

sifre db 5 dup(?) ; 5 karakterlik şifre tutacak alan

.code

main:

mov ax, @data

mov ds, ax

; Kullanıcıya mesaj

mov ah, 09h

lea dx, msg1

int 21h

; Giriş döngüsü (5 karakter)

mov cx, 5 ; 5 karakter

lea di, sifre ; sifre dizisinin adresi DI'ye

read\_loop:

; Karakteri al (gizli)

mov ah, 08h

int 21h

mov [di], al ; girilen karakteri belleğe kaydet

inc di

; '\*' yaz

mov dl, '\*' ; yıldız karakteri

mov ah, 02h

int 21h

loop read\_loop ; CX--, CX != 0 ise devam et

; Yeni satıra geç

mov ah, 09h

lea dx, msg2

int 21h

; Girilen şifreyi göster (kanıt)

lea si, sifre

mov cx, 5

print\_loop:

mov dl, [si]

mov ah, 02h

int 21h

inc si

loop print\_loop

; Programı bitir

mov ah, 4Ch

int 21h

end main

Döngü ile 1'den 5'e kadar sayıları toplama

.model small

.stack 100h

.data

sum db 0

.code

main:

mov ax, @data

mov ds, ax

mov cx, 5 ; döngü sayısı

mov al, 0 ; toplam AL'de tutulacak

mov bl, 1 ; başlama değeri

loop\_start:

add al, bl ; AL = AL + BL

inc bl ; BL = BL + 1

loop loop\_start ; CX = CX - 1, eğer CX != 0 ise döngüye devam

mov sum, al ; toplamı sum'a kaydet

mov ah, 4Ch

int 21h

end main

Sayı Karşılaştırma (If Else Mantığı)

.model small

.stack 100h

.data

a db 10

b db 20

greater db ?

.code

main:

mov ax, @data

mov ds, ax

mov al, a

cmp al, b ; a > b ?

ja isGreater ; eğer a > b ise isGreater etiketine git

; b büyüktür

mov greater, b

jmp endprog

isGreater:

mov greater, a

endprog:

mov ah, 4Ch

int 21h

end main

Örnek Program: Klavyeden Karakter Alıp Ekrana Yazdırma

.model small ; Bellek modeli: Kod ve veri segmenti küçük

.stack 100h ; Stack segmenti için 256 byte (100h) ayrılır

.data ; Veri segmenti başlangıcı

msg db 'Karakter girin: $' ; Mesaj (dolar işareti, DOS için string sonu belirtir)

.code ; Kod segmenti başlangıcı

main:

mov ax, @data ; Veri segmentinin adresini AX'e al

mov ds, ax ; AX içeriğini veri segmenti kaydedicisine (DS) aktar

; Ekrana mesaj yaz

mov ah, 09h ; DOS servis: 09h -> '$' ile biten string yaz

mov dx, offset msg ; msg dizisinin adresini DX'e al

int 21h ; DOS kesmesiyle çağır

; Karakter al

mov ah, 01h ; DOS servis: 01h -> klavyeden karakter oku

int 21h ; Sonuç AL'de döner

; Yazılan karakteri yeni satıra yaz

mov dl, 0Ah ; Line feed (LF)

mov ah, 02h ; DOS servis: 02h -> karakter yaz

int 21h

mov dl, 0Dh ; Carriage return (CR)

int 21h

mov dl, al ; AL'deki karakteri DL'ye taşı

int 21h ; AH zaten 02h

; Programı sonlandır

mov ah, 4Ch ; Programı DOS'a sonlandırmak için servis kodu

int 21h

end main ; Programın giriş noktası

İki Sayının Toplanması

.model small

.stack 100h

.data

num1 db 5

num2 db 7

result db ?

.code

main:

mov ax, @data

mov ds, ax

mov al, num1 ; AL = 5

add al, num2 ; AL = AL + num2 (5 + 7 = 12)

mov result, al ; sonucu result değişkenine aktar

; Programı sonlandır

mov ah, 4Ch

int 21h

end main

nseren1616@gmail.com