

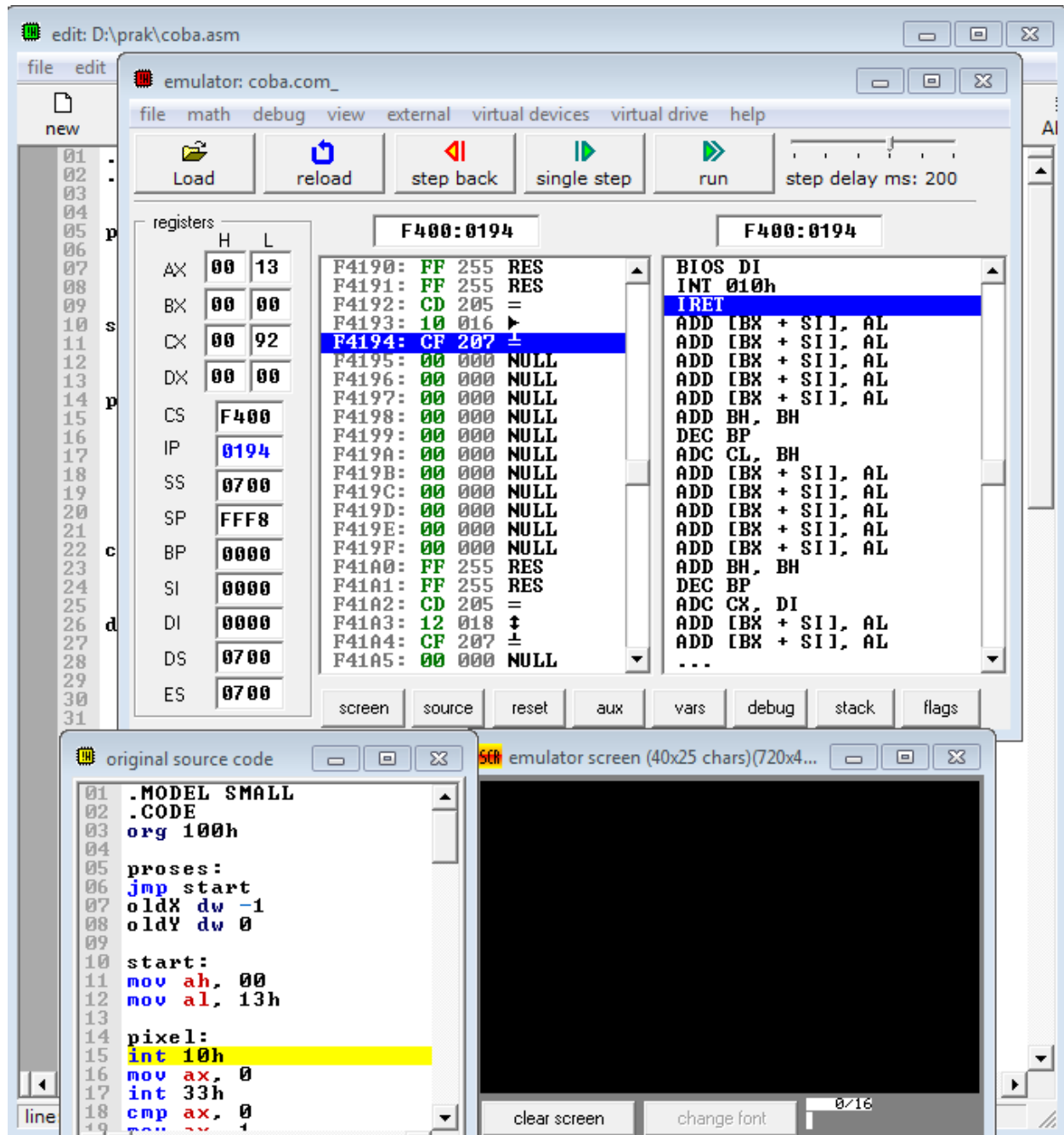
**LAPORAN PRAKTIKUM**  
**DASAR SISTEM KOMPUTER**



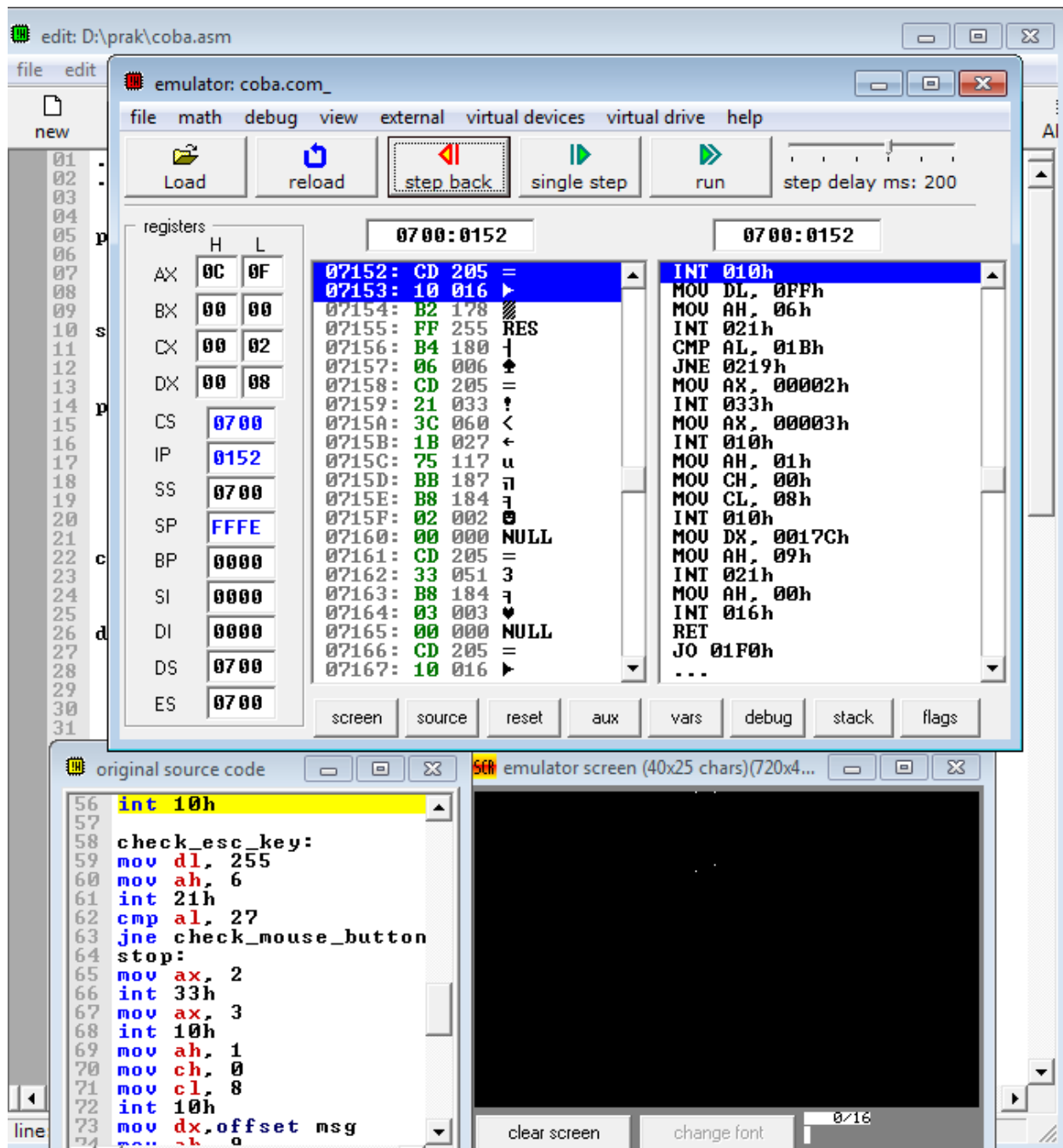
**DISUSUN OLEH:**  
**EKO RACHMAT SATRIYO (2100018142)**  
**JUM'AT 07.30-KELAS C**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**FAKULTAS TEKNOLOGI INDUSTRI**  
**UNIVERSITAS AHMAD DAHLAN**  
**DESEMBER 2021**

## POSTEST I

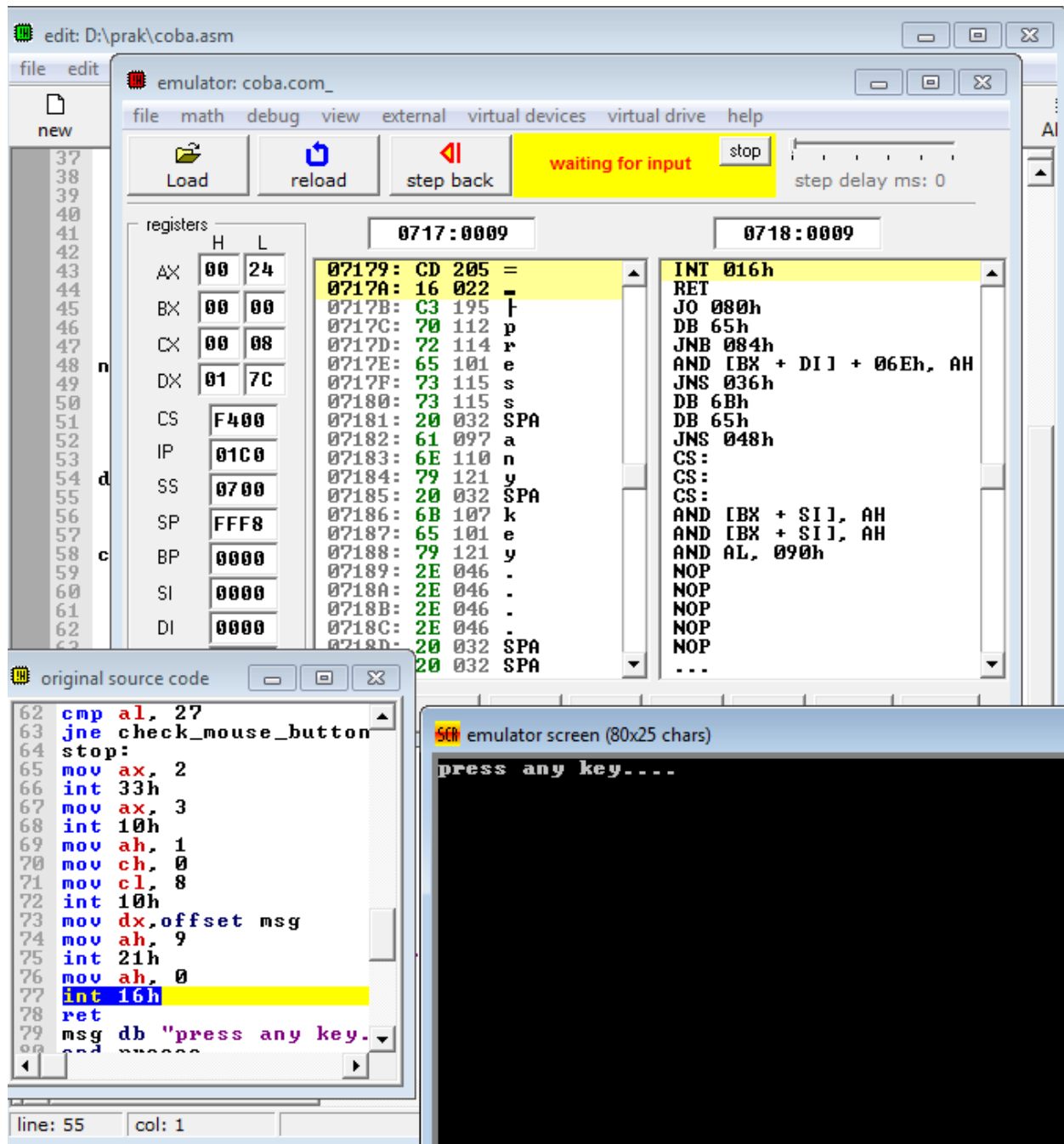


int 10h untuk mengubah pada tampilan layar(mengacu pada mov ah,00 dan mov al,13h)



Mulai menggambar pixel

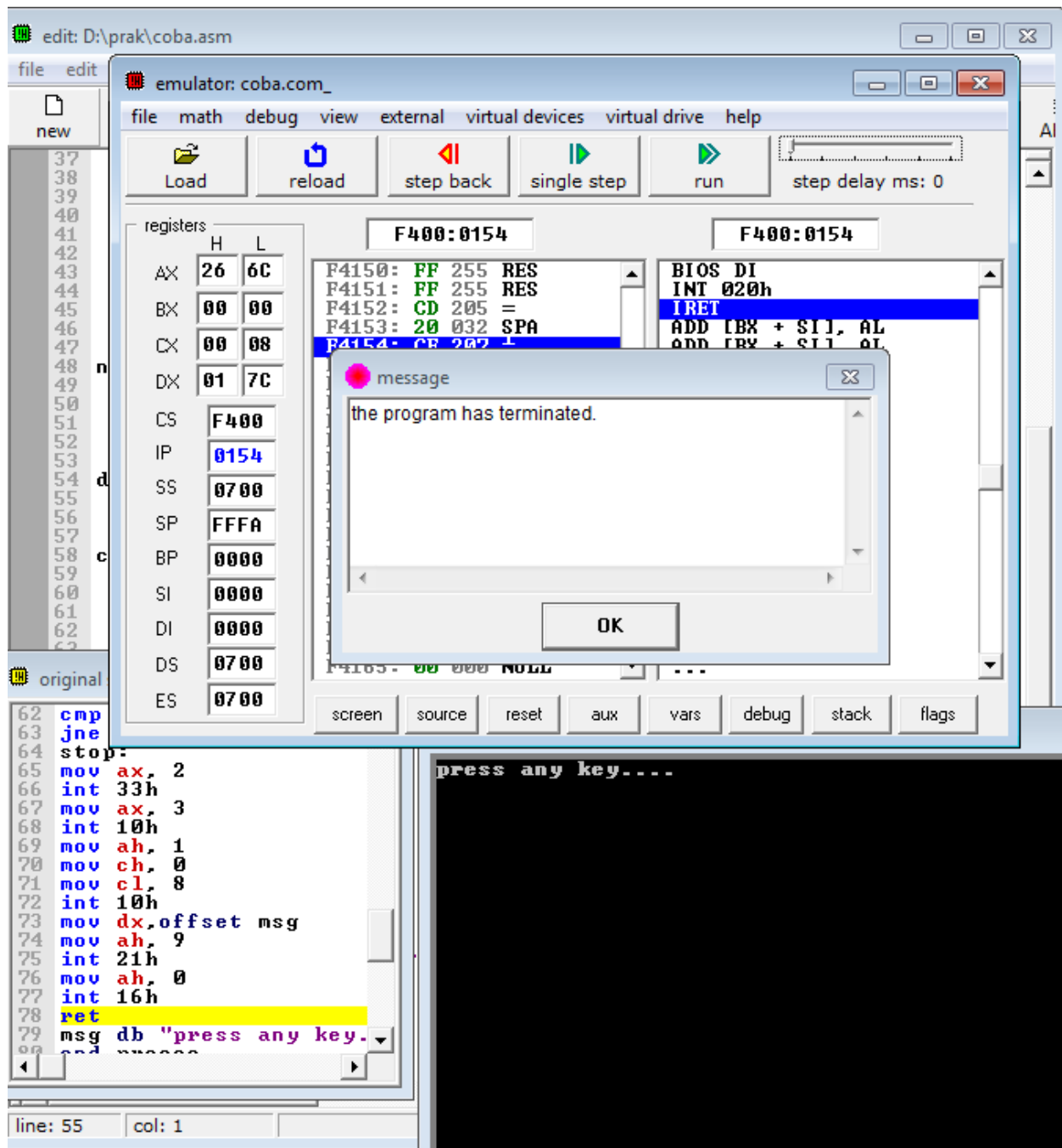
int 10h = untuk mengubah pada tampilan layar(mengacu mov ah, 0ch)



Ketika kita memencet "esc"

Akan tampil 80x25 char penyebabnya=

(berasal dari mov ax,3,int10h/baris ke 67 dan 68) dan apabila memencet suatu tombol,program akan berhenti



Memencet suatu tombol maka program akan berhenti/terminated

## POSTEST II

.MODEL SMALL

.CODE

org 100h           =menentukan alamat offset yang digunakan

proses:

jmp start           =lompat ke bagian start(dimulai dari mov ah,00)

oldX dw -1

oldY dw 0

start:

mov ah, 00        =mengubah ke mode grafik

mov al, 13h       =layar 256 warna,ukuran 320x200 pixel.

int 10h           =untuk mengubah pada tampilan layar(mengacu pada mov ah,00 dan mov al,13h)

mov ax, 0         =mereset mouse status saat ini

int 33h           =interupsi mouse

cmp ax, 0         =membandingkan nomor asli ke 0

mov ax, 1         =menampilkan kursor mouse

int 33h           =menginterupsi mouse(mengacu ke mov ax, 1)

check\_mouse\_button:

mov ax, 3         =mengembalikan posisi kursor mouse saat ini dan status tombol

int 33h           =menginterupsi mouse(mengacu ke mov ax, 3)

shr cx, 1         =menggeser 1 bit ke kanan pada register cx,hasil dari cx akan digandakan

cmp bx, 1         = membandingkan register bx dengan 1

jne xor\_cursor    =lompat ke xor\_cursor

mov al,1010b      =memasukkan nilai 10 ke register al,warna pixel

jmp draw\_pixel=lompat ke draw\_pixel

xor\_cursor:

cmp oldX,-1     =membandingkan oldx,dengan -1

je not\_required=lompat ke not\_required(apabila bernilai sama)

push cx         =memasukkan isi register cx ke dalam stack

push dx         = memasukkan isi register dx ke dalam stack

mov cx, oldX     =memasukkan oldX ke register cx

mov dx, oldY     =memasukkan oldY ke register dx

mov ah, 0dh     =memasukkan 13 ke register ah

int 10h         =mengubah tampilan pada layar

xor al,1111b     =melakukan bitwise eksklusif dan hasil disimpan ke register al,warna pixel

mov ah, 0ch     =memasukkan nilai 12 ke register ah,mengatur pixel

int 10h         = untuk mengubah pada tampilan layar(mengacu mov ah, 0ch)

pop dx           =mengeluarkan isi register dx dari stack

pop cx           = mengeluarkan isi register cx dari stack

not\_required:

mov ah, 0dh     =memasukkan nilai 13 ke register ah,mendapatkan pixel

int 10h         = untuk mengubah pada tampilan layar

xor al, 1111b    =warna pixel

mov oldx, cx     =menambahkan oldx dan cx,disimpan ke oldx

mov oldY, dx     =menambahkan oldY dan dx,disimpan ke oldx

draw\_pixel:

mov ah, 0ch     = memasukkan nilai 12 ke register ah,mengatur pixel

int 10h         = untuk mengubah pada tampilan layar(mengacu mov ah, 0ch)

check\_esc\_key:

mov dl, 255      =memasukkan nilai 255 ke register dl  
mov ah, 6        =memasukkan nilai 6 ke register ah  
int 21h          =menampilkan karakter  
cmp al, 27       =membandingkan register al dan 27  
jne check\_mouse\_button      =lompat ke check\_mouse\_button

stop:

mov ax, 2        =memasukkan nilai 2 ke register ax, menyembunyikan kursor mouse  
int 33h          = menginterupsi mouse(mengacu ke mov ax, 2)  
mov ax, 3        = mengembalikan posisi kursor mouse saat ini dan status tombol  
int 10h          = untuk mengubah pada tampilan layar(mengacu ke mov ax, 3)  
mov ah, 1        =menunjukkan bentuk kotak berkedip pada kursor text  
mov ch, 0        =memasukkan nilai 0 ke register ch  
mov cl, 8        =memasukkan nilai 8 ke register cl  
int 10h          = mengubah pada tampilan layar  
mov dx, offset msg=memasukkan alamat relatif(offset msg) ke register dx  
mov ah, 9        =memasukkan nilai 9 ke register ah  
int 21h          =mencetak karakter  
mov ah, 0        = memasukkan nilai 0(membaca tekan tombol)  
int 16h          =menginterupsi keyboard  
ret               =mengeluarkan alamat pengirim dari stack dan melanjutkan mengeksekusi  
alamat tersebut.



msg db "press any key.... \$"

=db digunakan untuk mendefinisikan pada variabel

\$ mengakhiri kalimat yang dicetak

end proses

=akhir dari proses