Müzeyyennur YILGIN

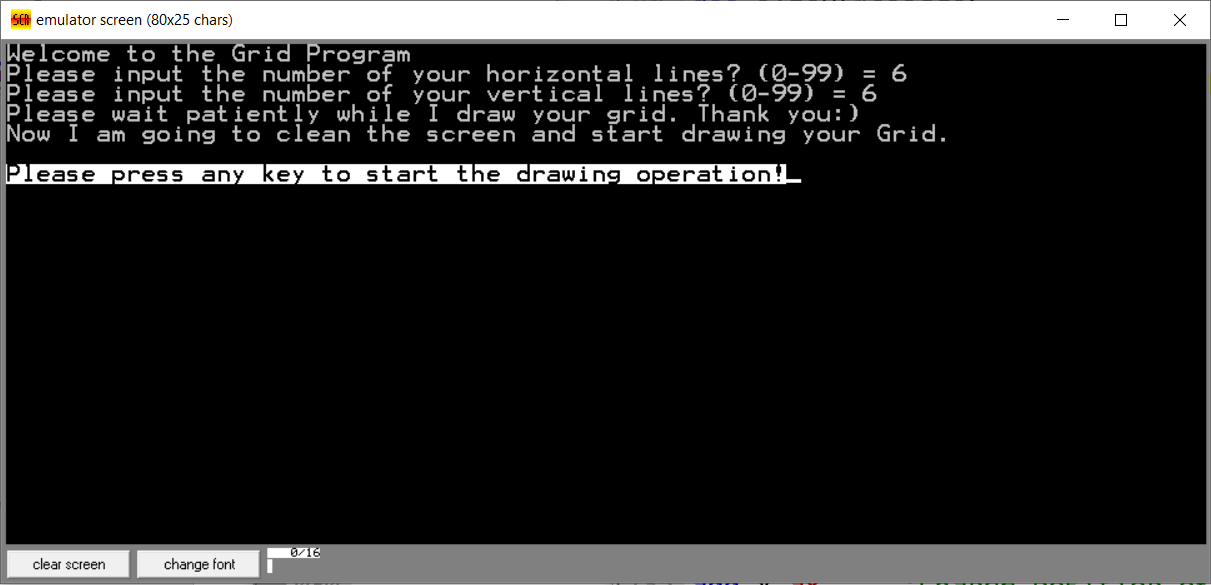
161180067

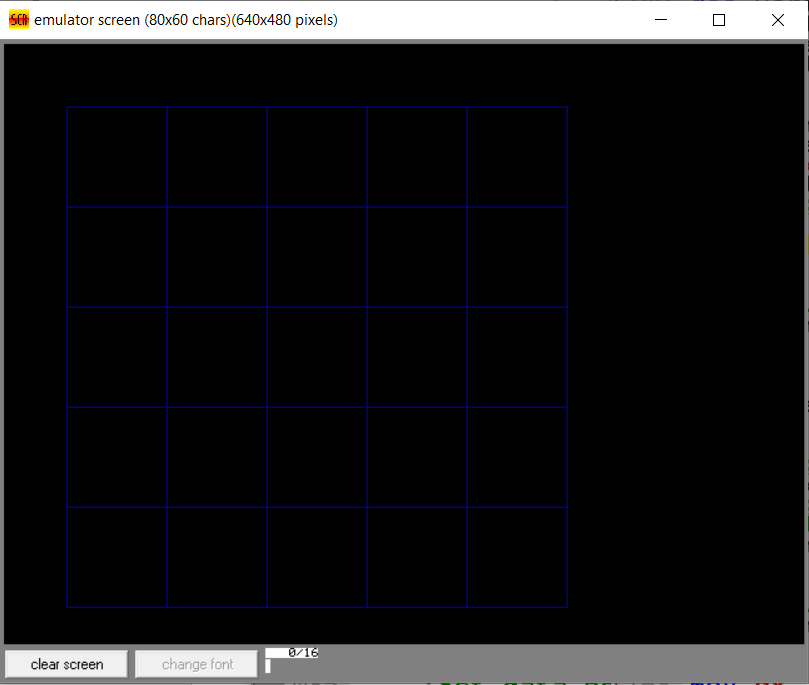
**GAZİ ÜNİVERSİTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ**

**CENG 318 – MICROPROCESSORS**

**MIDTERM**





.MODEL SMALL

.STACK 64h

.DATA

d db 10

number dw 0

sizeH dw 0

sizeV dw 0

x dw 50

y dw 50

sizeL dw 400

horizontalcounter dw 0

verticalcounter dw 0

newLine DB 0AH,0DH,"$" ;Newline character

message1 db 'Welcome to the Grid Program', '$'

message2 db 'Please input the number of your horizontal lines? (0-99) = ', '$'

message3 db 'Please input the number of your vertical lines? (0-99) = ', '$'

message4 db 'Please wait patiently while I draw your grid. Thank you :) ', '$'

message5 db 'Now I am going to clean the screen and start drawing your Grid. ', '$'

message6 db 'P','l','e','a','s','e',' ','p','r','e','s','s',' ','a','n','y',' ','k','e','y',' ','t','o',' ','s','t','a','r','t',' ','t','h','e',' ','d','r','a','w','i','n','g',' ','o','p','e','r','a','t','i','o','n','!', '$'

offsetOfmessage6 dw 0

sizeOfmessage6 db 51

cursorC db 0

err1 db 'Please enter number of horizontal lines in between (0,99)', '$'

err2 db 'Please enter number of vertical lines in between (0,99)', '$'

err3 db 'Please write a number!', '$'

.CODE

start:

;Clean screen

mov ax,0003h

int 10h

;Set data pointer

mov ax,@data

mov ds,ax

lea dx,d

;Print message1

lea dx,message1

mov ah,09h

int 21h

;Newline

lea dx,newLine

mov ah,9

int 21h

;Print message2

lea dx,message2

mov ah,09h

int 21h

;Take number of horizontal lines

call get2digits

mov ax,number ;Save value in number variable

mov horizontalcounter, ax ;Save value in accumulator to horizontalcounter variable

;Newline

lea dx,newLine

mov ah,9

int 21h

;Print message3

lea dx,message3

mov ah,9

int 21h

;Take number of vertical lines

call get2digits

mov ax,number ;Save value in number variable

mov verticalcounter, ax ;Save value in accumulator to verticalcounter variable

;Newline

lea dx,newLine

mov ah,9

int 21h

;Print message4

lea dx,message4

mov ah,09h

int 21h

;Newline

lea dx,newLine

mov ah,9

int 21h

;Print message5

lea dx,message5

mov ah,09h

int 21h

;Newline

lea dx,newLine

mov ah,9

int 21h

;Newline

lea dx,newLine

mov ah,9

int 21h

;Print message6 with blinking property

mov offsetOfmessage6,offset message6

dec offsetOfmessage6

;Loop to write char array to console

charLoop:

mov ah,09h

mov bx,offsetOfmessage6

inc bx

mov al,[bx]

mov offsetOfmessage6,bx

mov bh,0

mov bl,11110000b

mov cx,1

int 10h

call next ;Move cursor to the next column

dec sizeOfmessage6

jns charloop ;Continue to loop if counter is not 0

;Wait for a key press

mov ah,1

int 21h

;Clean screen

mov ax,0003h

int 10h

;Change display to 640x480

mov ax, 0012h

int 10h

mov ax, 0012h

int 10h

;Go skipH when horizontalcounter is 0

cmp horizontalcounter,0

jz skipH

;Go skipH1 when horizontalcounter is 1

cmp horizontalcounter,1

jz skipH1

mov ax,sizeL ;Get initial size of line

mov dx,0

dec horizontalcounter ;Count the gaps between lines

div horizontalcounter ;Compute new size of line between gaps

mov sizeH,ax ;Save computed size of line between gaps

inc horizontalcounter ;Reset the number of vertical lines

skipH1:

lh:

call horizontal ;Draw next horizontal line

mov ax,sizeH ;Get size of gap between vertical lines

add y,ax ;Change position of line

dec horizontalcounter ;Decrease the counter of horizontal lines

jnz lh ;Continue to loop if counter of horizontal lines is not 0

skipH:

mov x,50

mov y,50

;Go skipV when verticalcounter is 0

cmp verticalcounter,0

jz skipV

;Go skipV1 when verticalcounter is 1

cmp verticalcounter,1

jz skipV1

mov ax,sizeL ;Get initial size of line

mov dx,0

dec verticalcounter ;Count the gaps between lines

div verticalcounter ;Compute new size of line between gaps

mov sizeV,ax ;Save computed size of line between gaps

inc verticalcounter ;Reset the number of vertical lines

skipV1:

lv:

call vertical ;Draw next vertical line

mov ax,sizeV ;Get size of gap between vertical lines

add x,ax ;Change position of line

dec verticalcounter ;Decrease the counter of vertical lines

jnz lv ;Continue to loop if counter of vertical lines is not 0

skipV:

hlt

;Procedure to draw horizontal line

proc horizontal

mov ax, 0C01h ;Set properties of line

mov cx, x ;Get x coordinate

mov dx, y ;Get y coordinate

add cx,sizeL ;Go to next point as size of line

hl:

int 10h ;Draw point

dec CX ;Go to next point

cmp cx,x ;Control if x coordinate of the point is same with starting point of our x coordinate

jns hl ;If not continue to loop

ret ;Return

endp horizontal

;Procedure to draw vertical line

proc vertical

mov ax, 0C01h ;Set properties of line

mov cx, x ;Get x coordinate

mov dx, y ;Get y coordinate

add dx,sizeL ;Go to next point as size of line

vl:

int 10h ;Draw point

dec dx ;Go to next point

cmp dx,y ;Control if y coordinate of the point is same with starting point of our y coordinate

jns vl ;If not continue to loop

ret ;Return

endp vertical

;Procedure to get 2 digits number

proc get2digits

mov number,0

;Get first digit

mov ah,1

int 21h

;If Enter key pressed give error

cmp al,13

jz error3

sub al,48 ;To convert ascii code of the number character to numeric character

;If user entered a non-numeric character give error

js error3 ;If characters ascii code is smaller than 48 (smallest ascii code of a number is 48)

cmp al,10 ;Compare with 10

jns error3 ;If character is not 0,1,2,3,4,5,6,7,8 or 9

mov ah,0

mul d

;Add to number

mov ah,0

add number,ax

;Get second digit

mov ah,1

int 21h

;If Enter key pressed skip

cmp al,13

jz skip2ndD

sub al,48 ;To convert ascii code of the number character to numeric character

;If user entered a non-numeric character give error

js error3 ;If characters ascii code is smaller than 48 (smallest ascii code of a number is 48)

cmp al,10 ;Compare with 10

jns error3 ;If character is not 0,1,2,3,4,5,6,7,8 or 9

;Add to number

mov ah,0

add number,ax

ret ;return if we have only one digit

;Error if you dont write anything or write non-numeric character

error3:

;Newline

lea dx,newLine

mov ah,09h

int 21h

;Print err3

lea dx,err3

mov ah,09h

int 21h

;Wait for a key press

mov ah,1

int 21h

;Go to start

jmp start

skip2ndD:

mov ax,number ;Move first number to accumulator because we have only one digit

mov ah,0 ;Remove first digit

div d ;Divide value by 10 because we multiplied this by 10 before

mov ah,0 ;Remove the remainder

mov number,ax ;Move calculated number to number variable

ret ;Return

endp get2digits

;Procedure to go next column when row:6

proc next

mov dh,6

inc cursorC

mov dl,cursorC

mov bh,0

mov ah,2

int 10h ;Change cursor position

ret ;Return

endp next

end start