Emu-8086

5-3-24:

.model small

.stack 100h

.data

a dw 12

.code

main proc

mov ah,2

mov dl,'s'

int 21h

mov ah,4ch

int 21h

main endp

end main

5-3-24 part 2:

.MODEL SMALL

.STACK 100H

.CODE

MAIN PROC

MOV AH,2

MOV DL,'?'

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

5-3-24 part 2.1:

.MODEL SMALL

.STACK 100H

.CODE

MAIN PROC

;VALUE JA ASSIGN TAI SHOW KORBE

MOV AH,2 ;OUTPUT INSTRUCTION

MOV DL,'?' ;? PRINT HOBE

INT 21H

;USER INPUT

;NEW LINE ER JONNO

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

;NEW LINE CODE SES

MOV AH,1 ;USER INPUT INSTRUCTION ...SINGLE CHARACTER

INT 21H ;KEYBOARD THEKE INPUT DIBO

MOV BL,AL ;BL STORE/SHIFT KORBO

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,2 ;USER OUTPUT TA DEKHAR JONNO

MOV DL,BL ;BL A VALUE K DL A STRORE KORBO

INT 21H ;OUTPUT TA DEKHBO

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

12-3-24:

.model small

.stack 100h

;1 input initialize & 1 user input addition

.data

;variable initialize

msg db 'Hello World $' ;STING ER ENDING BUJHANOR JONNO & SIGN USE KORI

VAR1 DB 2

VAR2 DB ?

.code

main proc

MOV AX,@Data ;DATA SEG CODE INITIALIZE

MOV DS,AX

MOV AH,9

LEA DX,MSG ;STRING VARIABLE PRINT

;LOAD EFFECTIVE ADDRESS...VARIABLE ER ADDRESS K DIYE DIBE

INT 21H

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,1

INT 21H

MOV VAR2,AL

MOV BL,VAR2

ADD BL,VAR1

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

;output

MOV AH,2

MOV DL,BL

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

12-3-24 part 2:

.model small

.stack 100h

.data

A DB ?

B DB ?

.code

main proc

; MOV AX,@Data ;DATA SEG CODE INITIALIZE

; MOV DS,AX

;user 2 input addition

MOV AH,1

INT 21H

MOV A,AL

MOV BL,A

MOV AH,1

INT 21H

MOV B,AL

ADD BL,B

SUB BL,48

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,2

MOV DL,BL

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

12-3-24 part 3:

.model small

.stack 100h

;user 2 input addition

.data

;variable initialize

msg1 db 'Enter the 1st number: $'

msg2 db 'Enter the 2nd number: $'

msg3 db 'sum: $'

A DB ?

B DB ?

.code

main proc

MOV AX,@Data ;DATA SEG CODE INITIALIZE

MOV DS,AX

MOV AH,9

LEA DX,MSG1

INT 21H

MOV AH,1

INT 21H

MOV A,AL

MOV BL,A

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,9

LEA DX,MSG2

INT 21H

MOV AH,1

INT 21H

MOV B,AL

ADD BL,B

SUB BL,48

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,9

LEA DX,MSG3

INT 21H

MOV AH,2

MOV DL,BL

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

12-3-24 part 4:

.model small

.stack 100h

.data

str1 db 'start '

a db '2'

str2 db ' end $'

.code

main proc

MOV AX,@Data ;DATA SEG CODE INITIALIZE

MOV DS,AX

MOV AH,9

LEA DX,str1

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

;viva code

12-3-24 part 5:

.model small

.stack 100h

.code

main proc

MOV AH,2

MOV DL,'\*'

INT 21H

; MOV AH,2 ;na dileo hobe

MOV DL,'&'

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

;viva code

Add ex-4:

.model small

.stack 100h

.data

;variable initialize

msg db ? ;na dileo hobe .data part ta

.code

main proc

;? print

MOV AH,2

MOV DL,'?'

INT 21H

;user input

MOV AH,1

INT 21H

MOV CL,AL

;addition

MOV AH,1

INT 21H

ADD CL,AL

SUB CL,48 ;0 er ascii value 48..tai -48 decimal a anar jonno

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

;output

MOV AH,2

MOV DL,CL

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

Ex-8 main:

.model small

.stack 100h

.data

str1 DB 'THE SUM OF '

A DB ?

str2 DB ' AND '

B DB ?

str3 DB ' IS $'

.code

main proc

MOV AX,@Data ;DATA SEG CODE INITIALIZE

MOV DS,AX

MOV AH,2

MOV DL,'?'

INT 21H

MOV AH,1

INT 21H

MOV A,AL

MOV BL,A

MOV AH,1

INT 21H

MOV B,AL

ADD BL,B

SUB BL,48

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,9

LEA DX,str1

INT 21H

MOV AH,2

MOV DL,BL

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

Ex-8:

.model small

.stack 100h

;.data

;msg db ?

.code

main proc

;MOV AH,2

;MOV DL,'?'

;INT 21H

MOV AH,1

INT 21H

MOV BL,AL ;mov bl,al

MOV AH,1

INT 21H

MOV AH,AL ;MOV AL,BL ;mov bh,al

ADD BL,AH

SUB BL,48

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,2

MOV DL,BL

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

Ex-9:

.model small

.stack 100h

.data

msg db 'ENTER THREE INITIALG: $'

a db ?

b db ?

c db ?

.code

main proc

mov ax,@data ;initialize data segment

mov ds,ax

lea dx,msg

mov ah,9

int 21h

mov ah,1

int 21h

mov a,al

mov ah,1

int 21h

mov b,al

mov ah,1

int 21h

mov c,al

mov ah,2

mov dl,0dh ;caariage return

int 21h

mov dl,0ah ;new line

int 21h

mov ah,2

mov dl,a

int 21h

mov dl,0dh

int 21h

mov dl,0ah

int 21h

mov ah,2

mov dl,b

int 21h

mov dl,0dh

int 21h

mov dl,0ah

int 21h

mov ah,2

mov dl,c

int 21h

mov ah,4ch

int 21h

main endp

end main

Ex-11:

.model small

.stack 100h

.data

str db '\*\*\*\*\*\*\*\*\*\*$'

.code

main proc

mov ax,@data

mov ds,ax

lea dx,str

mov ah,9 ;print 10 times

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,str

mov ah,9

int 21h

mov ah,4ch

int 21h

main endp

end main

Ex-12 Ashraf:

.MODEL SMALL

.STACK 100H

.DATA

A DB 'ENTER FIRST DIGIT $'

B DB 'INPUT SECOND DIGIT $'

C DB 'RESULT $'

.CODE

MAIN PROC

MOV AX,@DATA

MOV DS,AX

MOV AH,9

LEA DX,A

INT 21H

MOV AH,1

INT 21H

MOV BL,AL

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,9

LEA DX,B

INT 21H

MOV AH,1

INT 21H

MOV BH,AL

;new line

MOV AH,2

MOV DL,0DH

INT 21H

MOV AH,2

MOV DL,0AH

INT 21H

MOV AH,9

LEA DX,C

INT 21H

ADD BL,BH ;B1 = B1+BH

SUB BL,48

MOV AH,2

MOV DL,BL

INT 21H

EXIT:

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN

Ex-12 nije:

.model small

.stack 100h

.data

msg1 db '\*\*\*\*\*\*\*\*\*\*\*$'

msg2 db '\* \*$'

msg3 db '\* $'

msg4 db ' \*$'

a db ?

b db ?

c db ?

.code

main proc

mov dl,'?'

mov ah,2

int 21h

mov ah,1

int 21h

mov a,al

;or mov bl,al

mov bl,a

mov ah,1

int 21h

mov b,al

; or mov cl,al

mov bh,b

mov ah,1

int 21h

mov c,al

;or mov bh,al

mov cl,c

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

mov ax,@data

mov ds,ax

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg3

mov ah,9

int 21h

mov dl,BL

mov ah,2 ;printing 1st scanned value

int 21h

mov dl,BH

int 21h ;printing 2nd scanned value

mov dl,CL ;printing 3rd scanned value

int 21h

lea dx,msg4

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,07h

int 21h

mov ah,4ch

int 21h

main endp

end main

Ex-12 pdf:

.model small

.stack 100h

.data

msg1 db '\*\*\*\*\*\*\*\*\*\*\*$'

msg2 db '\*\*\*\*$'

.code

main proc

mov dl,'?'

mov ah,2

int 21h

mov ah,1

int 21h

mov bl,al

mov ah,1

int 21h

mov cl,al

mov ah,1

int 21h

mov bh,al

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

mov ax,@data

mov ds,ax

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg2 ; printing less star to put the scanned value

mov ah,9

int 21h

mov dl,bl

mov ah,2 ;printing scanned value

int 21h

mov dl,cl

int 21h ;printing scanned value

mov dl,bh ;printing scanned value

int 21h

lea dx,msg2

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,0dh

int 21h

mov dl,0ah

int 21h

lea dx,msg1

mov ah,9

int 21h

mov ah,2

mov dl,07h

int 21h

mov ah,4ch

int 21h

main endp

end main

first practice:

.model small

.stack 100h

.code

main proc

mov ah,1

int 21h

mov bl,al

mov ah,2

mov dl,bl

int 21h

exit:

mov ah,4ch

int 21h

main endp

end main

video-3:

.MODEL SMALL

.STACK 100H

.DATA

MSG DB 3

MSG1 DB ?

.CODE

MAIN PROC

;DATA SEGMENT INITIALIZE

MOV AX,@DATA

MOV DS,AX ; SES

MOV AH,2

ADD MSG,48

MOV DL,MSG

INT 21H

MOV AH,1

INT 21H

MOV MSG1,AL

MOV AH,2

MOV DL,10

INT 21H

MOV DL,13

INT 21H

MOV AH,2

MOV DL,MSG1

INT 21H

MOV AH,4CH

INT 21H

MAIN ENDP

END MAIN