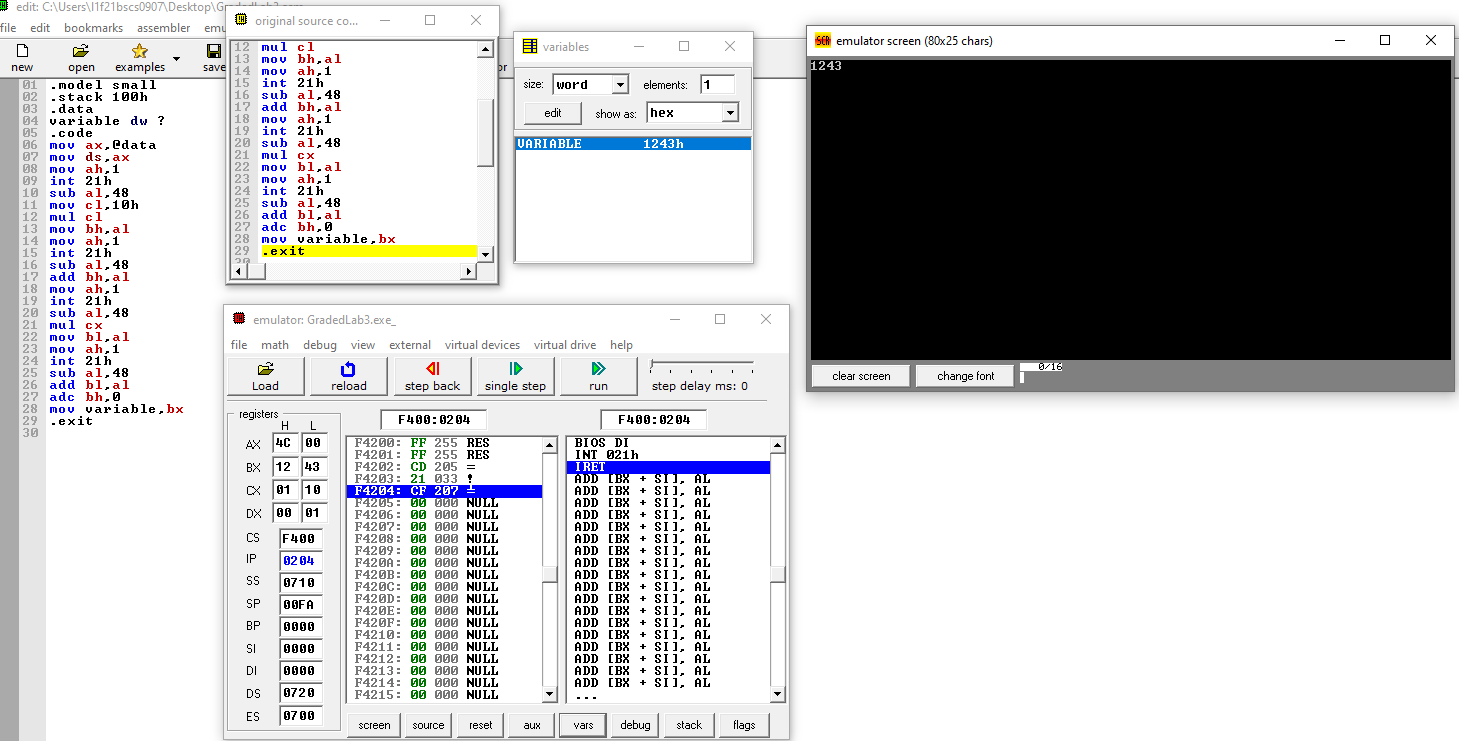
TASK # 1



.model small

.stack 100h

.data

variable dw ?

.code

mov ax,@data

mov ds,ax

mov ah,1

int 21h

sub al,48

mov cl,10h

mul cl

mov bh,al

mov ah,1

int 21h

sub al,48

add bh,al

mov ah,1

int 21h

sub al,48

mul cx

mov bl,al

mov ah,1

int 21h

sub al,48

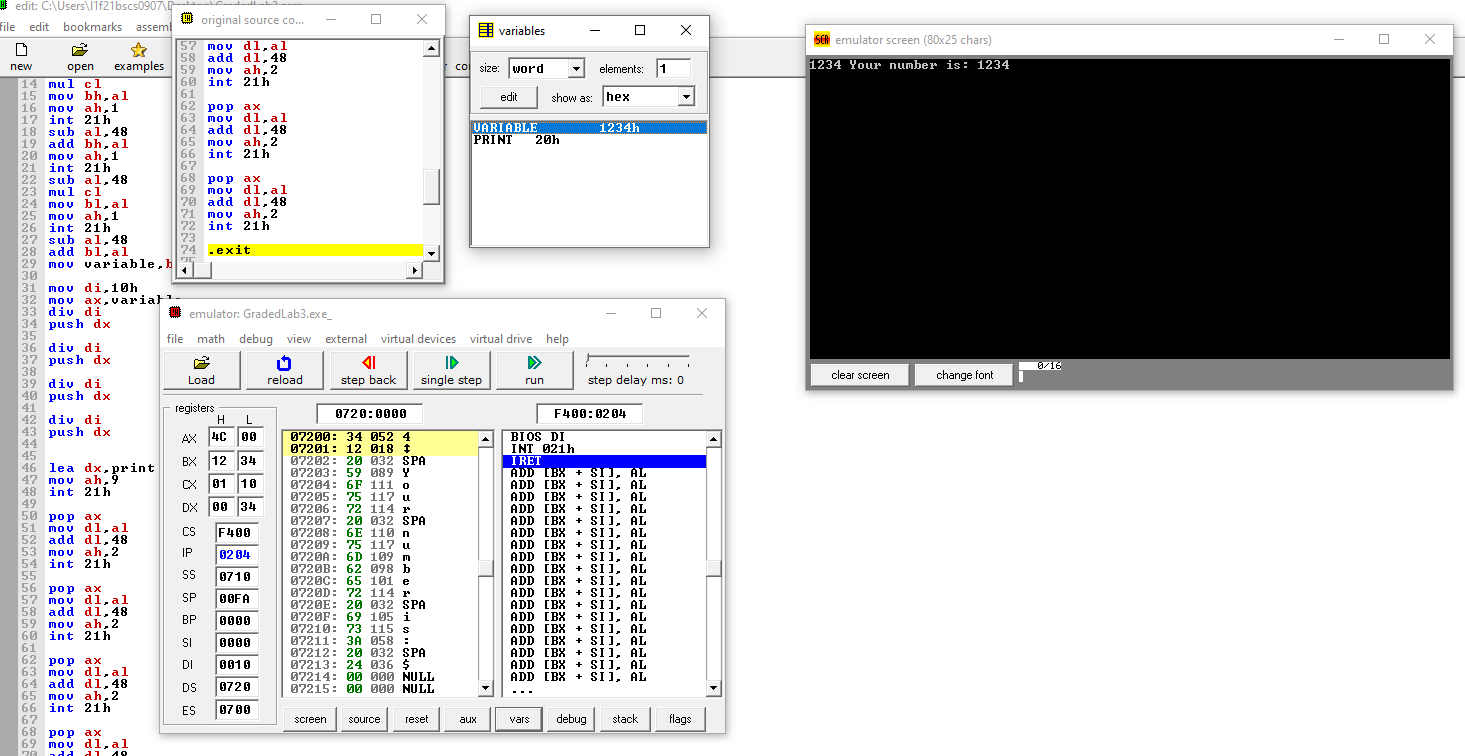
add bl,al

adc bh,0

mov variable,bx

.exit

TASK # 2



.model small

.stack 100h

.data

variable dw ?

print db " Your number is : $"

.code

mov ax,@data

mov ds,ax

mov ah,1

int 21h

sub al,48

mov cl,10h

mul cl

mov bh,al

mov ah,1

int 21h

sub al,48

add bh,al

mov ah,1

int 21h

sub al,48

mul cl

mov bl,al

mov ah,1

int 21h

sub al,48

add bl,al

mov variable,bx

mov di,10h

mov ax,variable

div di

push dx

div di

push dx

div di

push dx

div di

push dx

lea dx,print

mov ah,9

int 21h

pop ax

mov dl,al

add dl,48

mov ah,2

int 21h

pop ax

mov dl,al

add dl,48

mov ah,2

int 21h

pop ax

mov dl,al

add dl,48

mov ah,2

int 21h

pop ax

mov dl,al

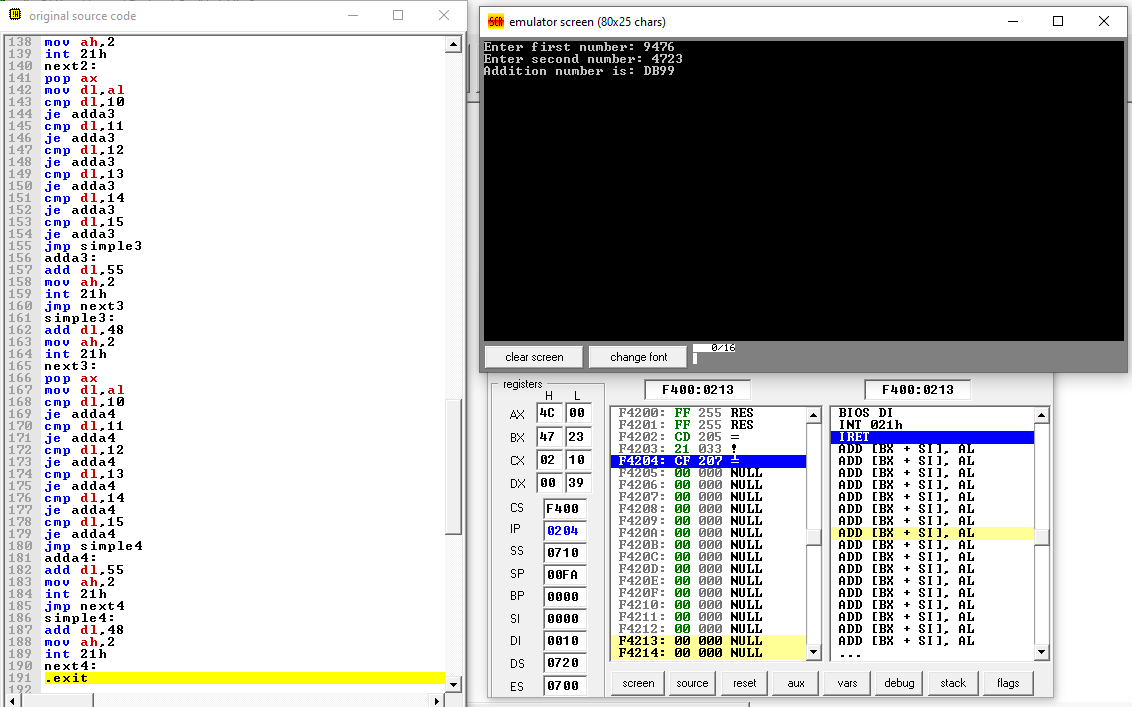
add dl,48

mov ah,2

int 21h

.exit

TASK # 3

.model small

.stack 100h

.data

variable1 dw ?

variable2 dw ?

ans dw ?

print1 db "Enter first number : $"

print2 db "Enter second number : $"

print3 db "Addition number is : $"

.code

mov ax,@data

mov ds,ax

lea dx,print1

mov ah,9

int 21h

mov ah,1

int 21h

sub al,48

mov cl,10h

mul cl

mov bh,al

mov ah,1

int 21h

sub al,48

add bh,al

mov ah,1

int 21h

sub al,48

mul cx

mov bl,al

mov ah,1

int 21h

sub al,48

add bl,al

adc bh,0

mov variable1,bx

mov dl,10d

mov ah,2

int 21h

mov dl,13d

mov ah,2

int 21h

lea dx,print2

mov ah,9

int 21h

mov ah,1

int 21h

sub al,48

mov cl,10h

mul cl

mov bh,al

mov ah,1

int 21h

sub al,48

add bh,al

mov ah,1

int 21h

sub al,48

mul cx

mov bl,al

mov ah,1

int 21h

sub al,48

add bl,al

adc bh,0

mov variable2,bx

mov ax,variable1

mov bx,variable2

add ah,bh

adc al,bl

mov ans,ax

mov di,10h

mov ax,ans

div di

push dx

div di

push dx

div di

push dx

div di

push dx

mov dl,10d

mov ah,2

int 21h

mov dl,13d

mov ah,2

int 21h

lea dx,print3

mov ah,9

int 21h

pop ax

mov dl,al

cmp dl,10

je adda1

cmp dl,11

je adda1

cmp dl,12

je adda1

cmp dl,13

je adda1

cmp dl,14

je adda1

cmp dl,15

je adda1

jmp simple1

adda1:

add dl,55

mov ah,2

int 21h

jmp next1

simple1:

add dl,48

mov ah,2

int 21h

next1:

pop ax

mov dl,al

cmp dl,10

je adda2

cmp dl,11

je adda2

cmp dl,12

je adda2

cmp dl,13

je adda2

cmp dl,14

je adda2

cmp dl,15

je adda2

jmp simple2

adda2:

add dl,55

mov ah,2

int 21h

jmp next2

simple2:

add dl,48

mov ah,2

int 21h

next2:

pop ax

mov dl,al

cmp dl,10

je adda3

cmp dl,11

je adda3

cmp dl,12

je adda3

cmp dl,13

je adda3

cmp dl,14

je adda3

cmp dl,15

je adda3

jmp simple3

adda3:

add dl,55

mov ah,2

int 21h

jmp next3

simple3:

add dl,48

mov ah,2

int 21h

next3:

pop ax

mov dl,al

cmp dl,10

je adda4

cmp dl,11

je adda4

cmp dl,12

je adda4

cmp dl,13

je adda4

cmp dl,14

je adda4

cmp dl,15

je adda4

jmp simple4

adda4:

add dl,55

mov ah,2

int 21h

jmp next4

simple4:

add dl,48

mov ah,2

int 21h

next4:

.exit