

این برنامه از ۳ proc تشکیل شده است

۱ - proc main نخست با استفاده از تابع 09 پیغام mes1 را نمایش و سپس با استفاده از تابع 0ah عدد ۲ رقمی با علامت را از کاربر از طریق صفحه کلید دریافت میکند.

۲ - در مرحله بعد از proc atoi برای تبدیل یک عدد صحیح علامت دار دورقمی اسکی ذخیره شده در offset DI (برای مثال 12- یا 42+) به معادل عدد صحیح مکمل ۲ و ذخیره آن در AL استفاده میشود. سپس عدد بدست آمده در AL در مکان a1 ذخیره میشود.

۳ - سپس با استفاده از proc disp پیغام mes3 با استفاده از تابع 09 نمایش داده میشود و عدد صحیح ۲ رقمی مکمل دو در AL تبدیل به شکل ascii (شکل اولیه آن) و در offset di ذخیره و سپس با تابع 09 نمایش داده میشود.

۴ - مراحل فوق برای عدد دوم تکرار میشود، با این تفاوت که نخست پیغام mes2 نمایش داده میشود و عدد صحیح بدست آمده در a2 ذخیره میگردد.

```
stk segment
```

```
dw 32 dup (?)
```

```
stk ends
```

```
dts segment
```

```
max db 4
```

```
len db ?
```

```
str1 db 3 dup (?)
```

```
a1 db ?
```

```
a2 db ?
```

```
sta1 db 'a1= $'
```

```
sta2 db 'a2= $'
```

```
mes1 db 10,13,' Please enter first number $'
```

```
mes2 db 10,13,' Please enter second number $'
```

```
mes3 db 10,13,' your number is: $'
```

```
dts ends
```

```
procs segment
```

```
assume cs:procs
```

```
atoi proc far
```

```
; converts signed 2 digit number ( +12 or -78 ) in ascii format with offset in SI to integer- final result in AL
```

```
mov bl,10
```

```
mov al,[si+1]
and al,0fh ; converts the ascii digit to integer
mul bl
mov cl,[si+2] ; second digit
and cl,0fh
add al,cl
```

```
cmp byte ptr [si], '-'
    jne L1
```

```
neg al
```

```
L1:    ret
```

```
atoi endp
```

```
disp  proc far
```

```
; converts signed 2 digit integer in al to ascii and stores in offset di
```

```
mov byte ptr [di], '+'
```

```
test al,80h ; testing sign bit; zf=1 if sign bit=zero
```

```
jz pos
```

```
mov byte ptr [di], '-'
```

```
neg al
```

```
pos:  mov bl,10
```

```
xor ah,ah ; make ah 0
```

```
div bl
```

```
or ax,3030h ; convert al and ah to ascii code of numbers
```

```
mov byte ptr [di+2],ah
```

```
mov byte ptr [di+1],al
```

```
;display the number on screen
```

```
mov ah,09
```

```
mov dx, offset mes3
```

```
int 21h
```

```
mov ah,09
```

```
mov dx,di
```

```
int 21h
```

```

        ret
disp    endp
procs   ends

clds    segment
        assume cs:clds,ds:dts,ss:stk
main    proc far
        mov ax, seg dts
        mov ds, ax
        mov es,ax
; display message 1 - read the first number from key board

        mov ah,09
        mov dx, offset mes1
        int 21h

        mov ah, 0ah
        mov dx, offset max
        int 21h
        mov si, offset str1

; call atoi to convert the number ( in ascii format ) in to integer

        call atoi

; store converted number in a1
        mov a1,a1

; convert the number a1 back to ascii and store in sta1 and display

        mov di, offset sta1
        call disp

; display message 2 - read the second number from key board

```

```
mov ah,09
mov dx, offset mes2
int 21h
```

```
    mov ah, 0ah
    mov dx, offset max
    int 21h
mov si, offset str1
```

; call atoi to convert the number (in ascii format) in to integer

```
    call atoi
    mov a2,a1
```

; convert the number a2 back to ascii and store in sta2 and display

```
mov di,offset sta2
    call disp
```

```
    mov ah,4ch
    int 21h
    ret
```

```
main  endp
```

```
cds   ends
```

```
    end main
```