.model small

.stack 100h

.data

giris\_msg db 'Bir sayi giriniz (1-255): $'

sonuc\_cift db 13,10, 'Cift Sayilarin Toplami: $'

sonuc\_tek db 13,10, 'Tek Sayilarin Toplami : $'

sayi\_str db 4 dup(0)

cift\_str db 6 dup(0), '$'

tek\_str db 6 dup(0), '$'

girilen\_sayi dw ?

cift\_toplam dw 0

tek\_toplam dw 0

.code

start:

mov ax, @data

mov ds, ax

*; Kullanıcıdan sayı iste*

mov ah, 09h

mov dx, offset giris\_msg

int 21h

*; Sayı girişi*

mov byte ptr [sayi\_str], 3

lea dx, sayi\_str

mov ah, 0Ah

int 21h

*; ASCII -> Sayıya çevir*

xor ax, ax

xor bx, bx

lea si, sayi\_str + 2

mov cl, byte ptr [sayi\_str+1]

conv\_loop:

mov bl, byte ptr [si]

sub bl, '0'

mov dx, ax

mov ax, 10

mul dx

add ax, bx

inc si

loop conv\_loop

mov girilen\_sayi, ax

*; 1'den girilen sayıya kadar döngü*

mov cx, 1

dongu:

mov ax, cx

mov dx, 0

mov bx, 2

div bx *; ax / 2 → dx = kalan*

cmp dx, 0

jne tek\_sayi

*; Çift ise*

add cift\_toplam, cx

jmp devam

tek\_sayi:

add tek\_toplam, cx

devam:

inc cx

mov ax, girilen\_sayi

cmp cx, ax

jbe dongu

*; Çift toplamı yazdır*

mov ax, cift\_toplam

lea di, cift\_str

call int\_to\_str

mov ah, 09h

mov dx, offset sonuc\_cift

int 21h

mov ah, 09h

mov dx, offset cift\_str

int 21h

*; Tek toplamı yazdır*

mov ax, tek\_toplam

lea di, tek\_str

call int\_to\_str

mov ah, 09h

mov dx, offset sonuc\_tek

int 21h

mov ah, 09h

mov dx, offset tek\_str

int 21h

*; Programdan çık*

mov ah, 4Ch

int 21h

*; ----------------------------------------*

*; Sayıyı stringe çevirme (AX → DI)*

int\_to\_str:

push ax

push bx

push cx

push dx

xor cx, cx

mov bx, 10

dondur:

xor dx, dx

div bx

add dl, '0'

push dx

inc cx

test ax, ax

jnz dondur

yazdir:

pop dx

mov byte ptr [di], dl

inc di

loop yazdir

mov byte ptr [di], '$'

pop dx

pop cx

pop bx

pop ax

ret

end start