section .text

calculateOne:

mov ebp,caracterbusqueda ;tomar primer caracter

mov edx,caracteres ;tomar primer caracter de los 100

mov di,0 ;inicio de contador en 0

mov word [count],0 ;inicio de contador en -1

cant:

mov al,[edx] ;tomo el caracter actual

cmp al,[ebp] ;comparar 100 dígitos con el primero a buscar

je incremento

cmp di,100 ;comparar si el contador ha llegado a 100

je sigue

inc edx ;apunto a la siguiente posición

inc di ;aumento el contador

jmp cant

ret

incremento:

inc word [count] ;incrementar contador

inc di ;incrementar contador

inc edx ;apunto a la siguiente posición

jmp cant

calculateTwo:

mov ebp,caracterbusqueda ;tomar primer caracter

mov edx,caracteres ;tomar primer caracter de los 100

mov di,0 ;inicio de contador en 0

mov word [count],0 ;inicio de contador en 0

cant2:

mov al,[edx] ;tomo el caracter actual

cmp al,[ebp] ;comparar 100 dígitos con el primero a buscar

je incremento2

cmp al,[ebp+2] ;comparar 100 dígitos con el segundo a buscar

je incremento2

cmp di,100 ;comparar si el contador ha llegado a 100

je sigue

inc edx ;apuntar a la siguiente posición

inc di ;incremento el contador

jmp cant2

ret

incremento2:

inc word [count] ;incrementar contador

inc di ;incrementar contador

inc edx ;incrementar contador

jmp cant2

global \_start

\_start:

mov eax,4     ;sys\_wite

mov ebx,1     ;To stdout

mov ecx,mensaje     ;'Input some data

mov edx,mensaje\_size ;size del mensaje

int 80h ;llamada al sistema operativo

mov eax,3     ;sys\_read. Read what user inputs

mov ebx,0     ;From stdin

mov ecx,caracteres  ;buffer de caracteres para buscar

mov edx,100 ;size de caracteres para buscar

int 80h ;llamada al sistema operativo

mov eax,4     ;sys\_wite

mov ebx,1     ;To stdout

mov ecx,mensajebusqueda     ;mensaje

mov edx,mensajebusqueda\_size ;size del mensaje de búsqueda

int 80h ;llamada al sistema operativo

mov eax,3     ;sys\_read.

mov ebx,0     ;From stdin

mov ecx,caracterbusqueda  ;caracteres a buscar

mov edx,3 ;size de caracteres de búsqueda

int 80h ;llamada al sistema operativo

mov ebp,caracterbusqueda ;muevo el primer caracter a un registro

mov cl,'+' ;muevo el + a un registro

cmp [ebp+1],cl ;pregunto si el segundo caracter es +

jne calculateOne

jmp calculateTwo

sigue:

mov eax,4     ;sys\_wite

mov ebx,1     ;To stdout

mov ecx,mensajesigue    ;imprimo el mensaje de resultado

mov edx,mensajesigue\_size ;size del mensaje de resultado

int 80h ;llamada al sistema operativo

mov ebx,[count] ;muevo el contador de caracteres a un registro

add ebx, 0x30 ;le sumo 0x30 al registro para que me del número en decimal

mov [count],ebx ;muevo el resultado a un buffer

int 80h ;llamada al sistema operativo

mov eax,4     ;sys\_wite

mov ebx,1     ;To stdout

mov ecx,count     ;imprimo el resultado

mov edx,1 ;size del resultado

int 80h ;llamada al sistema operativo

mov eax,4     ;sys\_wite

mov ebx,1     ;To stdout

mov ecx,mensajesalto    ;salto de línea

mov edx,mensajesalto\_size ;size del salto de línea

int 80h ;llamada al sistema operativo

salir:

mov eax,1 ;salir del sistema

mov ebx,0 ;salir del sistema

int 80h ;llamada al sistema operativo

section .bss

caracterbusqueda resb 3 ;buffer de caracteres para buscar

caracteres resb 100 ;buffer para 100 bytes de caracteres

count resw 1 ;buffer para contar letras

section .data

mensaje: db 'Digite 100 caracteres o menos:',0 ;string

mensaje\_size: equ $-mensaje ;size del string

mensajebusqueda: db 'Digite la letra o suma de letras para buscar:',0 ;string

mensajebusqueda\_size: equ $-mensajebusqueda ;size del string

mensajesigue: db 'Resultado: ',0 ;string

mensajesigue\_size: equ $-mensajesigue ;size del string

mensajesalto: db 0ah,0 ;string

mensajesalto\_size: equ $-mensajesalto ;size del string

resultado: db 'Hay: ','$' ;string

resultado\_size: equ $-resultado ;size del string