; Tecnològico de Costa Rica

; Escuela de Computaciòn

; Arquitectura de Computadores IC-3101

; Segundo semestre, 2019

; Profesor Esteban Arias

; Tarea #2 - Conversiòn de texto a nùmero

; Estudiante Marian Soza Hidalgo

; Carnè 2018123188

; 09/09/2019

%include "io.mac"

.DATA

number\_prompt db "Please type your number: ",0

out\_msg db "Your number is: ",0

invalid\_msg db "Invalid number",0

banderita db "Wenas",0

espacio db " ",0

.UDATA

in\_number resb 13

variable resb 32

.CODE

.STARTUP

PutStr number\_prompt ; request number in string

GetStr in\_number,13

mov EBX,in\_number

mov ESI,0 ;Check sign

mov EDI,0 ;Check dot

mov ECX,0 ;Result

push 1

mov EBP,0

process\_beg:

mov AL,[EBX] ; move the char to AL

cmp AL,0 ; if it is the NULL

je done

cmp AL,45

je negative\_sgn

cmp AL,43

je validate\_numb

cmp AL,48

jb invalid\_numb

cmp AL,57

ja invalid\_numb

inc EBX

inc EBP

validate\_numb:

mov AL,[EBX]

cmp AL,0 ; if it is the NULL

je pre\_convert ;salir

cmp AL,46

je float\_point

cmp AL,48

jb invalid\_numb

cmp AL,57

ja invalid\_numb

inc EBX

inc EBP

jmp validate\_numb

pre\_convert:

cmp EDI,1

dec EBX

je convert\_fl

mov EDI,1 ;exponente

mov EDX,10

jmp convert\_int

convert\_int:

mov EAX, 0

mov AL,[EBX]

sub AL,'0'

mov EDX,EDI

mul EDX

add ECX,EAX

mov EAX,EDI

mov EDX,10

mul EDX

mov EDI,EAX

cmp EBP,1

je print\_res

dec EBX

dec EBP

jmp convert\_int

convert\_fl:

jmp done

float\_point:

inc EDI

inc EBX

mov AL,[EBX]

cmp AL,48

jb invalid\_numb

cmp AL,57

ja invalid\_numb

jmp validate\_numb

print\_res:

cmp ESI, 1

je c2

mov [variable], ECX

PutLInt [variable]

nwln

mov EAX,0

mov [variable], EAX

mov EAX,ECX

mov EBX,2

mov ESI,31

mov EDI,0

jmp make\_bin

c2:

not ECX

inc ECX

mov [variable],ECX

PutLInt [variable]

nwln

mov EAX,ECX

mov EBX,2

mov ESI,31

mov EDI, 0

jmp make\_bin

make\_bin:

mov EDX, 0

div EBX

add EDX,'0'

mov [variable+ESI],DL

dec ESI

cmp ESI, -1

je pre\_sp

jmp make\_bin

pre\_sp:

mov EDX, 0

mov EAX, 0

print\_bin:

cmp EDX, 32

je done

mov BL, [variable+EDX]

PutCh BL

inc EAX

inc EDX

cmp EAX, 4

jne print\_bin

PutStr espacio

mov EAX, 0

jmp print\_bin

negative\_sgn:

mov ESI,1

inc EBX

mov AL,[EBX]

cmp AL,48

jb invalid\_numb

cmp AL,57

ja invalid\_numb

jmp validate\_numb

invalid\_numb:

PutStr invalid\_msg

jmp done

done:

nwln

.EXIT

to\_end:

inc EBX

cmp AL,0 ; if it is the NULL

jne to\_end

dec EBX

mov AL,[EBX]

ret