**3]write an alp to display two strings on the screen using EQU directory**

**Input:**

section .data

LF EQU 10

NULL EQU 0

SYS\_WRITE EQU 4

SYS\_EXIT EQU 1

STDOUT EQU 1

msg1 db 'Hello, Assembly!', LF, NULL msg1\_len EQU $ - msg1 - 1

msg2 db 'Using EQU directive', LF, NULL msg2\_len EQU $ - msg2 - 1

section .text global \_start

\_start:

mov eax, SYS\_WRITE mov ebx, STDOUT mov ecx, msg1 mov edx, msg1\_len int 0x80

mov eax, SYS\_WRITE mov ebx, STDOUT mov ecx, msg2 mov edx, msg2\_len int 0x80

mov eax, SYS\_EXIT xor ebx, ebx int 0x80

**Ouput :**



**4]Write an ALP to replace a word in a given string**

**Input –**

section .data

string db 'Hello World', 0

len equ $ - string

old db 'Hello'

new db 'Hi '

newline db 10, 0

section .text

global \_start

\_start:

mov eax, 4

mov ebx, 1

mov ecx, string

mov edx, len

int 0x80

mov esi, 0

mov ecx, 5

compare:

mov al, [string + esi]

mov bl, [old + esi]

cmp al, bl

jne exit

inc esi

loop compare

replace:

mov esi, 0

mov ecx, 5

rep\_loop:

mov al, [new + esi]

mov [string + esi], al

inc esi

loop rep\_loop

exit:

mov eax, 4

mov ebx, 1

mov ecx, string

mov edx, len

int 0x80

mov eax, 4

mov ebx, 1

mov ecx, newline

mov edx, 1

int 0x80

mov eax, 1

xor ebx, ebx

int 0x80

**OUTPUT**



-+