**PROGRAMS**

1. Write an ALP to display ‘Hello World’ on the screen.

section .data

hello db 'hello world', 10

helloLen equ $-hello

section .text

global \_start

\_start:

mov eax, 4

mov ebx, 1

mov ecx, hello

mov edx, helloLen

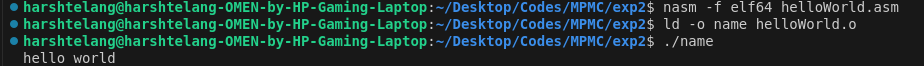
int 80h

mov eax, 1

mov ebx, 0

int 80h

Output:



1. **Write an ALP to display 9 stars on screen using times directive.**

section .data

stars TIMES 9 db '\*'

starLen equ $-stars

section .text

global \_start

\_start:

mov eax, 4

mov ebx, 1

mov ecx, stars

mov edx, starLen

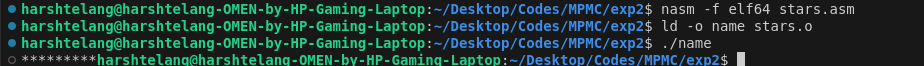
int 80h

mov eax, 1

mov ebx, 0

int 80h

Output:



1. **Write an ALP to display two strings on screen.**

section .data

string1 db 'Hello ', 0

string2 db 'World!', 0

string1Len equ $-string1

string2Len equ $-string2

section .text

global \_start

\_start:

mov eax, 4

mov ebx, 1

mov ecx, string1

mov edx, string1Len

int 80h

mov ecx, string2

mov edx, string2Len

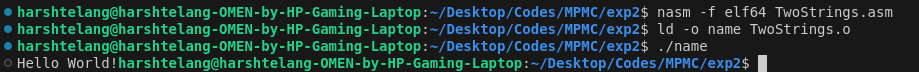
int 80h

mov eax, 1

mov ebx, 0

int 80h

Output:



1. **Write an ALP to replace a given word in string.**

section .data

name db 'abc def ,'

namelen equ $-name

section .text

global \_start

\_start:

; Print initial string

mov eax, 4

mov ebx, 1

mov ecx, name

mov edx, namelen

int 80h

; Replace "abc" with "xyz"

mov dword [name], dword 'xyz '

; Print modified string

mov eax, 4

mov ebx, 1

mov ecx, name

mov edx, namelen

int 80h

mov eax, 1

mov ebx, 0

int 80h

Output: 

1. **Write an ALP that reads a number from keyboard and displays it on the screen.**

section .data

number resb 5

section .text

global \_start

\_start:

mov eax, 3

mov ebx, 0

mov ecx, number

mov edx, 5

int 80h

mov eax, 4

mov ebx, 1

mov ecx, number

mov edx, 5

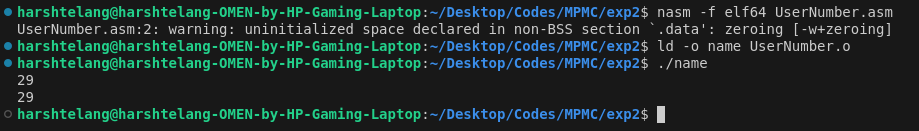
int 80h

mov eax, 1

mov ebx, 0

int 80h

Output:



1. **Write an ALP to print your name on screen.**

section .data

name resb 10

section .text

global \_start

\_start:

mov eax, 3

mov ebx, 0

mov ecx, name

mov edx, 10

int 80h

mov eax, 4

mov ebx, 1

mov ecx, name

mov edx, 10

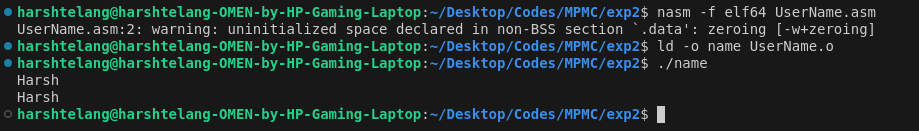
int 80h

mov eax, 1

mov ebx, 0

int 80h

Output:



1. **Write an ALP to accept number and string and display it on the screen.**

section .data

name resb 10

number resb 5

section .text

global \_start

\_start:

mov eax, 3

mov ebx, 0

mov ecx, name

mov edx, 10

int 80h

mov ecx, number

mov edx, 5

int 80h

mov eax, 4

mov ebx, 1

mov ecx, name

mov edx, 10

int 80h

mov ecx, number

int 80h

mov eax, 1

mov ebx, 0

int 80h

Output:

