**ML Assignment**

**Name:** Jagruti Ravindra Patil

**Roll No:** 2089

**Batch:** S4 Comp

**Assignment 4:**

**Code:**

section .data

num1 db 4h

num2 db 2h

msg db "Enter 1(+).Add",10

msg\_len equ $-msg

msg1 db "Enter 2(-).Subtract",10 msg1\_len equ $-msg1

msg2 db "Enter 3(\*).Multiply",10 msg2\_len equ $-msg2

msg3 db "Enter 4(/).Divide",10 msg3\_len equ $-msg3

msg4 db "Enter your choice:(+,-,\*,/) ",10 msg4\_len equ $-msg4

msg5 db "Result: "

msg5\_len equ $-msg5

msg6 db 10,"Quotient: "

msg6\_len equ $-msg6

msg7 db 10,"Remainder: "

msg7\_len equ $-msg7

msg8 db "Error: Enter correct choice" msg8\_len equ $-msg8

section .bss

chc resb 2

res resb 4

section .text

global \_start

\_start:

mov rax, 1

mov rdi, 1

mov rsi, msg

mov rdx, msg\_len syscall

mov rax, 1

mov rdi, 1

mov rsi, msg1 mov rdx, msg1\_len syscall

mov rax, 1

mov rdi, 1

mov rsi, msg2 mov rdx, msg2\_len syscall

mov rax, 1

mov rdi, 1

mov rsi, msg3 mov rdx, msg3\_len syscall

mov rax, 1

mov rdi, 1

mov rsi, msg4 mov rdx, msg4\_len syscall

mov rax, 0

mov rdi, 0

mov rsi, chc

mov rdx, 2

syscall

cmp byte[chc],'+' je Addition

cmp byte[chc],'-' je Subtraction cmp byte[chc],'\*' je Multiplication cmp byte[chc],'/' je Division

jmp lw

lw:

mov rax, 1

mov rdi, 1

mov rsi, msg8 mov rdx, msg8\_len syscall

mov rax, 60

mov rdi, 0

syscall

Addition:

call add1 mov rax, 60

mov rdi, 0

syscall

add1:

mov al,[num1] mov bl,[num2] mov ah,00

mov bh,00

add ax,bx

mov bx,ax

call disp

ret

Subtraction:

call sub1 mov rax, 60

mov rdi, 0

syscall

sub1:

mov al,[num1] mov bl,[num2]

mov ah,00

mov bh,00

sub ax,bx

mov bx,ax

call disp

ret

Multiplication:

call multi1 mov rax, 60

mov rdi, 0

syscall

multi1:

mov al,[num1] mov bl,[num2]

mov ah,00

mov bh,00

mul bx

mov bx,ax

call disp

ret

Division:

call div1 mov rax, 60

mov rdi, 0

syscall

div1:

mov al,[num1] mov bl,[num2] mov dx,00

mov ah,00

mov bh,00

div bx

push ax

push dx

mov rax, 1

mov rdi, 1

mov rsi, msg7 mov rdx, msg7\_len syscall

pop dx

mov bx,dx

call disp

mov rax, 1

mov rdi, 1

mov rsi, msg6 mov rdx, msg6\_len syscall

pop ax

mov bx,ax

call disp

ret

disp:

mov rdi,res

mov cx,16

d1:

rol bx,04

mov al,bl

and al,0fh

cmp al,09h

jg up2

add al,30h

jmp down

up2:

add al,37h

down:

mov [rdi],al inc rdi

dec cx

jnz d1

mov rax, 1

mov rdi, 1

mov rsi, msg5 mov rdx, msg5\_len syscall

mov rax, 1

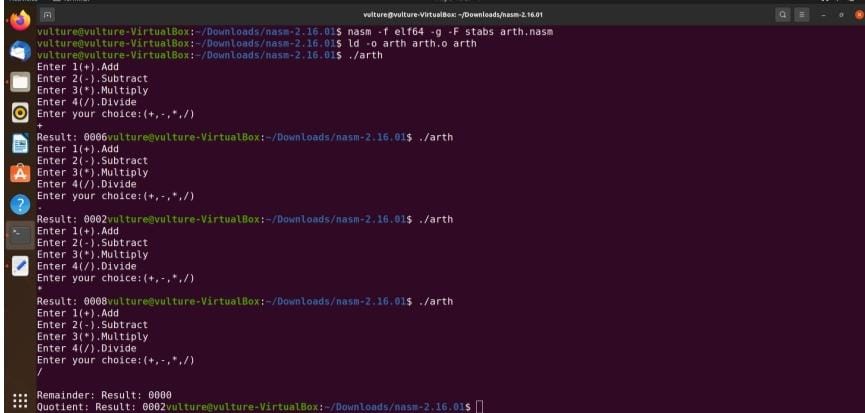
mov rdi, 1

mov rsi,res

mov rdx, 4

syscall

ret

**Output:**