Name: **Abdullah Khan.**

Lab Course Title: **Computer Organization and Architecture Lab.**

Semester: **5th Semester (Fall 2022).**

REG #. **20PWCSE1916.**

Assignment

Computer Organization and Architecture Lab

Department of Computer system Engineering

Submitted to:

Dr. Amaad Khalil

Calculator in MIPS assembly language

Write a MIPS assembly language program to implement a simple calculator that can perform addition, subtraction, multiplication, and division operations. The calculator should prompt the user to enter two operands and the operation they want to perform. After performing the operation, the result should be displayed on the console.

```
.data
i1: .asciiz "Enter first number: "
info: .asciiz "Enter operator: "
i2: .asciiz "\nEnter second number: "
res: .asciiz "\nYour result is: "
quotient: .asciiz "\nThe quotient is: "
remainder: .asciiz "\nThe remainder is: "

space: .asciiz "\n\n"

.text
main:

a:   # Label

la $a0, i1  # Load address of input 1 in $a0
# -------------- Print String i1 -----------------
li $v0, 4
syscall
# ------------------------------------------

# -------------- Read 1st Number  --------------------
li $v0, 5
syscall
# ------------------------------------------
move $t0, $v0

la $a0, info
# -------------- Print String info -----------------
li $v0, 4
syscall
# ------------------------------------------

# -------------- Read Operator  --------------------
li $v0, 12
syscall
# ------------------------------------------
move $t2, $v0
```

```
la $a0, i2  # Load address of input 2 in $a0
# -------------- Print String i2 -----------------
li $v0, 4
syscall
# ------------------------------------------

# -------------- Read 2nd Number  --------------------
li $v0, 5
syscall
# ------------------------------------------
move $t1, $v0

li $t3, 43 # t3 is a temporary register used for beq instruction
beq $t2, $t3, addition # If t3 is + (Ascii: 43) jump to addtion

li $t3, 45
beq $t2, $t3, subtraction # If t3 is - (Ascii: 45) jump to subtraction

li $t3, 42
beq $t2, $t3, multiplication # If t3 is * (Ascii: 42) jump to multiplication

li $t3, 47
beq $t2, $t3, division # If t3 is / (Ascii: 47) jump to division

j cont

    addition:
add $t4, $t0, $t1
j cont

    subtraction:
sub $t4, $t0, $t1
j cont

    multiplication:
mul $t4, $t0, $t1
j cont

    division:
div $t4, $t0, $t1
j divis

    divis:
la $a0, quotient
```

```
# -------------- Print Quotient String -----------------
li $v0, 4
syscall
# ------------------------------------------------

mflo $a0
# -------------- Print Int -----------------
li $v0, 1
syscall
# ------------------------------------------------

la $a0, remainder
# -------------- Print Remainder String -----------------
li $v0, 4
syscall
# ------------------------------------------------

mfhi $a0
# -------------- Print Int -----------------
li $v0, 1
syscall
# ------------------------------------------------

  cont:
la $a0, res
# -------------- Print String -----------------
li $v0, 4
syscall
# ------------------------------------------------

move $a0, $t4
# -------------- Print Int -----------------
li $v0, 1
syscall
# ------------------------------------------------

la $a0, space
# -------------- Print String i1 -----------------
li $v0, 4
syscall
# ------------------------------------------------

# -------------- Jump -----------------
j a
```

```
li $v0, 10 # syscall code 10 is for exit.
syscall

# end of real_calculator.asm
```