# 4.1 WAP to perform addition & subtraction and print the result

li $t0,9 # t0 = 9

li $t1,4 # t1 = 4

add $t2,$t0,$t1 # t2 = t0 + t1

move $a0,$t2 # a0 = t2

li $v0,1 # v0 = 1 for printing an integer

syscall

li $a0,10 # a0 = 10

li $v0,11 # v0 = 11 for printing a new line

syscall

# li $a0,32 for space

# li $v0,11

# syscall

sub $t3,$t0,$t1 # t3 = t0 - t1

move $a0,$t3 # a = t3

li $v0,1 # v0 = 1 for printing an integer

syscall

li $v0,10 # end / exit

syscall

# 4.2 WAP to perform addition & subtraction on user generated values and print the result

li $v0,5 # read an integer from user

syscall

move $t0,$v0 # t0 = v0 getting the input

li $v0,5 # read an integer from user

syscall

move $t1,$v0 # t1 = v0 getting the input

add $t2,$t0,$t1 # t2 = t0 + t1

move $a0,$t2 # a0 = t2

li $v0,1 # v0 = 1, prints an integer

syscall

li $a0,10 # a0 = 10, prints a new line

li $v0,11 # v0 = 11, prints a new line

syscall

# li $a0,32 for space

# li $v0,11

# syscall

sub $t3,$t0,$t1 # t3 = t0 - t1

move $a0,$t3 # a0 = t3

li $v0,1 # v0 = 1, prints and integer

syscall

li $v0,10 # end / exit

syscall

# 4.3 WAP to perform addition & subtraction & multiply & divide on user generated values and print the result

li $v0,5 # read an integer from user

syscall

move $t0,$v0 # t0 = v0 getting the input

li $v0,5 # read an integer from user

syscall

move $t1,$v0 # t0 = v0 getting the input

add $t2,$t0,$t1 # t2 = t0 + t1

move $a0,$t2 # a0 = t2

li $v0,1 # v0 = 1, prints an integer

syscall

li $a0,10 # a0 = 10, prints a new line

li $v0,11 # v0 = 11, prints a new line

syscall

# li $a0,32 for space

# li $v0,11

# syscall

sub $t3,$t0,$t1 # t3 = t0 - t1

move $a0,$t3 # a0 = t3

li $v0,1 # v0 = 1, prints and integer

syscall

li $a0,10 # a0 = 10, prints a new line

li $v0,11 # v0 = 11, prints a new line

syscall

mul $t4, $t0, $t1 # t4 = t0 \* t1

mflo $a0 # move low to a0

li $v0,1 # print an integer

syscall

li $a0,10 # a0 = 10, prints a new line

li $v0,11 # v0 = 11, prints a new line

syscall

div $t4, $t0, $t1 # t4 = t0 / t1

mflo $a0 # move low to a0

li $v0,1 # print an integer

syscall

li $a0,32 # a0 = 32, prints space

li $v0,11 # v0 = 11, prints space

syscall

mfhi $a0 # move high to a0

li $v0,1 # print an integer

syscall

li $v0,10 # end / exit

syscall