**Sample Code Date- 3/5/2021**

1. Construct a solid rectangle with ‘\*’ using MIPS assembly language. Get the length and breadth from the user . Hint[Use jump/conditional branching ] . Eg: length – 6, breadth-2.

\*\*\*\*\*\*

\*\*\*\*\*\*

**Answer**

#Hint : Nested for loop

.data

msg1: .asciiz "Enter length of recatangle "

msg2: .asciiz "Enter breadth of rectangle "

lf: .asciiz "\n"

star: .asciiz "\*"

.text

.globl main

main:

# Print msg1

li $v0,4 # print\_string syscall code = 4

la $a0, msg1

syscall

# Get length from user and save

li $v0,5 # read\_int syscall code = 5

syscall

move $t0,$v0 # syscall results returned in $v0 = 3

# Print msg2

li $v0,4 # print\_string syscall code = 4

la $a0, msg2

syscall

# Get breadth from user and save

li $v0,5 # read\_int syscall code = 5

syscall

move $t1,$v0 # syscall results returned in $v0 = 6

# Initialize registers

li $t2, 0 # initialize t2 = 0

beq $t0,$zero,exit

beq $t1,$zero,exit

# Main loop body

loop1: beq $t2, $t1, exit

addi $t2, $t2, 1

li $t3, 0 # initialize sum

li $v0,4 # print\_string syscall code = 4

la $a0, lf

syscall

loop2: beq $t3, $t0, loop1

addi $t3, $t3, 1 # increment length

li $v0, 4

la $a0, star

syscall

j loop2

# Print newline

exit: li $v0,4 # print\_string syscall code = 4

la $a0, lf

syscall

li $v0,10 # exit

syscall

**Lab Sheet 6**

Qn 3 # Count number of 1s in a 32 Bit Number

#$t0 = input number

#$t1 = counter register

.data

input: .asciiz "Enter Number: "

output: .asciiz "Number of 1s counted: "

n: .word 0

cnt:.word 0

.text

.globl main

main:

#Prompt User for Number

li $v0, 4 #Display text

la $a0, input #Input

syscall # string display

#Get user Input

li $v0, 5 #load

la $t0, n #load number into $t0

syscall #display integer

#Store user input in $t0 to do function

move $t0, $v0 #Move value from $v0 to $t0

la $t1, cnt #load address of counter

lw $t1, 0($t1) #load counter to t1

ANDLOOP:

andi $t2, $t0, 1 #and user input($t0) with 1 and store in $t2

beq $t2, 1, loop #if t2 equals 0 branch

srl $t0, $t0, 1 #shift user input($t0) to right 1 position logically

j ANDLOOP

loop:

add $t1, $t1, 1 #add 1 to the counter variable

srl $t0, $t0, 1 #shift $to to the right 1 position logically

beq $t0,$zero,dis #If $t0 equals 0 , display the result

j ANDLOOP

dis:

li $v0, 4 #load text stored in v0

la $a0, output #print text from address a0

syscall

la $a0, ($t1) #load the address of the counter to a0

li $v0, 1 #load integer stored in v0

syscall #print final integer

li $v0,10 # exit

syscall