TASK 01

```
.data
string1: .asciiz "Enter the Table Number = "
string2: .asciiz "\n"
a: .word 1
b: .word 1
number: .word 1
s1: .asciiz "*"
s2: .asciiz "="

.text
lw $t0, a
lw $t1, b
lw $t2, number
      #string1 (TABLE NUMBER)
li $v0, 4
la $a0 , string1
syscall
      #input
li $v0, 5
syscall
move $t0 , $v0

      #LOOP
loop:
      # compare
beq $t1, 11, Exit

# Multiply
mul $t2, $t0, $t1

      # (a)
move $a0, $t0
li $v0, 1
syscall
      # ( * sign )
li $v0, 4
la $a0 , s1
syscall
      # (b)
move $a0, $t1
li $v0, 1
syscall
      # ( = sign )
li $v0, 4
la $a0 , s2
syscall
      # NUMBER OUTPUT
move $a0, $t2
```

```
li $v0,1
syscall
      #string 2 (NEWLINE)
li $v0, 4
la $a0 , string2
syscall

            # ( b++)
add $t1, $t1, 1
j loop

Exit:
li $v0, 10
syscall
```

```
Enter the Table Number = 4
4*1=4
4*2=8
4*3=12
4*4=16
4*5=20
4*6=24
4*7=28
4*8=32
4*9=36
4*10=40
```

TASK 02

```
.data
string1: .asciiz "Enter a positive Integer = "
string2: .asciiz "Sum = "
num: .word 0
count: .word 0
sum: .word 0

.text
lw $t0, num
lw $t1, count
lw $t2, sum
      #String1 Enter Integer
li $v0, 4
la $a0, string1
syscall
      #Integer Input
li $v0, 5
syscall
move $t0 , $v0
```

```
loop:
      #Condition
bge $t1, $t0, exit

      #Increment
add $t1, $t1, 1

      # Sum Calculation
add $t2, $t2, $t1

j loop
exit:

#string SUM =
li $v0, 4
la $a0, string2
syscall

#Print Sum
move $a0, $t2
li $v0,1

syscall
li $v0, 10
syscall
```

```
Enter a positive Integer = 5
Sum = 15
-- program is finished running --
```

TASK 03

```
.data
string1:.asciiz "\n"

i:.word 0

.text

lw $t0, i
      # i=15 intialization
add $t0, $t0, 15
```

```
# OUTPUT Starting from 15
move $a0, $t0
li $v0,1
syscall

#newline
li $v0,4
la $a0, string1
syscall

loop:
      # i>0
blez  $t0, exit

      #i-=2
sub $t0, $t0, 2

      # after -2
move $a0, $t0
li $v0,1
syscall

#newline
li $v0,4
la $a0, string1
syscall

j loop
exit:
li $v0, 10
syscall
```

```
15
13
11
9
7
5
3
1
```