**Example # 1:**

**Solution**

.data

mess1:.asciiz "Enter Number 1 : "

mess2:.asciiz "Enter Number 2 : "

res:.asciiz "Your result is : "

.text

.globl main

main:

li $v0,4

la $a0,mess1

syscall

li $v0,5

syscall

move $t0,$v0

li $v0,4

la $a0,mess2

syscall

li $v0,5

syscall

move $t1,$v0

bgtz $t0,true

sub $t2 $t0,$t1 #Subtract The Number

li $v0,4

la $a0,res

syscall

move $a0,$t2

li $v0,1

syscall

b exit

true:

add $t2 $t0,$t1 #Add The Number

li $v0,4

la $a0,res

syscall

move $a0,$t2

li $v0,1

Text

Description automatically generatedsyscall

b exit

exit:

li $v0,10

syscall

**Output**

Table

Description automatically generated

**Example # 2:**

**Solution**

.data

mess1:.asciiz "Enter Number 1 : "

mess2:.asciiz "Enter Number 2 : "

res:.asciiz "Your result is : "

pr1:.asciiz "A is less Than B "

pr2:.asciiz "B is less Than A "

.text

.globl main

main:

li $v0,4

la $a0,mess1

syscall

li $v0,5

syscall

move $t0,$v0

li $v0,4

la $a0,mess2

syscall

li $v0,5

syscall

move $t1,$v0

blt $t0,$t1,true

li $v0,4 #asd

la $a0,res

syscall

li $v0,4

la $a0,pr2

syscall

b exit

true:

li $v0,4

la $a0,res

**Text

Description automatically generated**syscall

li $v0,4

la $a0,pr1

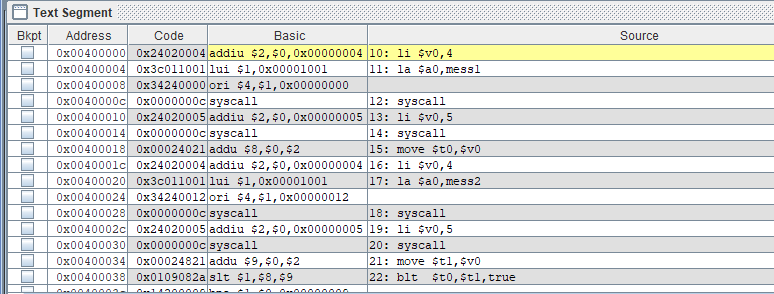
syscall

b exit

exit:

li $v0,10

**Output**

****

**Task No. 1:** Write a program in MIPS assembly language that takes input from user and print whether the input is greater or less than 10 and also shift input left and right 4 bits.

**Solution:**

.data

mees1:.asciiz"Enter Number : "

prom1:.asciiz"Numer Is Less then 10 "

prom2:.asciiz"number Is Greater Then 10"

prom3:.asciiz"Number Is Equal To 10"

.text

.globl main

main:

addi $t1,$zero,10

li $v0,4

la $a0,mees1

syscall

li $v0,5

syscall

move $t0,$v0

slt $t2,$t0,$t1

beq $t0,10,equal

beq $t2,0,if

li $v0,4

la $a0,prom1

syscall

b exit

equal:

li $v0,4

la $a0,prom3

syscall

b exit

if:

li $v0,4

la $a0,prom2

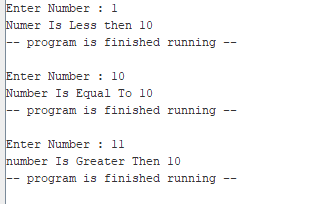
syscall

b exit

exit:

li $v0,10

syscall

**Output**

**Task No. 2:** Write a program in MIPS assembly language that takes input from user shift input left and right 4 bits.

**Solution:**

**Left Shift**

.data

mess1:.asciiz"Enter Number : "

mess2:.asciiz"Enter Number You Want To shift Left : "

result:.asciiz"Result is : "

.text

main:

li $v0,4

la $a0,mess1

syscall

li $v0,5

syscall

move $t0,$v0

li $v0,4

la $a0,mess2

syscall

li $v0,5

syscall

move $t1,$v0

sllv $t3,$t0,$t1

li $v0,4

la $a0,result

syscall

move $a0,$t3

li $v0,1

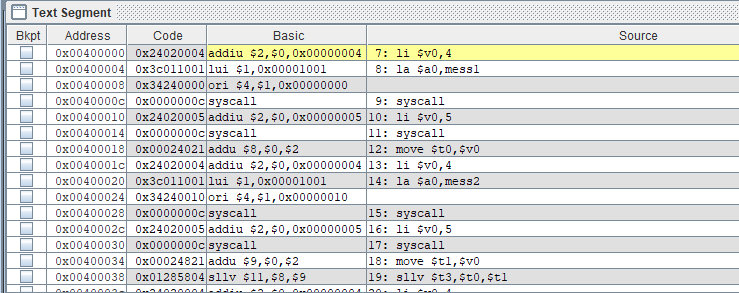
**Graphical user interface, text, application, Word

Description automatically generated**syscall

li $v0,10

syscall

**Output**

****

**Right Shift**

**Solution**

.data

mess1:.asciiz"Enter Number : "

mess2:.asciiz"Enter Number You Want To shift Left : "

result:.asciiz"Result is : "

.text

main:

li $v0,4

la $a0,mess1

syscall

li $v0,5

syscall

move $t0,$v0

li $v0,4

la $a0,mess2

syscall

li $v0,5

syscall

move $t1,$v0

srlv $t3,$t0,$t1

li $v0,4

la $a0,result

syscall

move $a0,$t3

li $v0,1

syscall

li $v0,10

syscall

**Graphical user interface, text, application

Description automatically generatedOutput:**

**Table

Description automatically generated**