**Logical Instruction, Use of lo and hi Registers through, Multiplication and Division & Character Manipulation**

**LAB # 05**



**Fall 2022**

**CSE304L Computer Organization & Architecture**

Submitted by:

Danyal Khursheed (20pwcse1959)

Maaz Habib (20pwcse1952)

Hamid (20pwcse1969)

Class Section: **C**

Student Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_

Submitted to:

**Dr. Amaad Khalil**

**Department of Computer Systems Engineering**

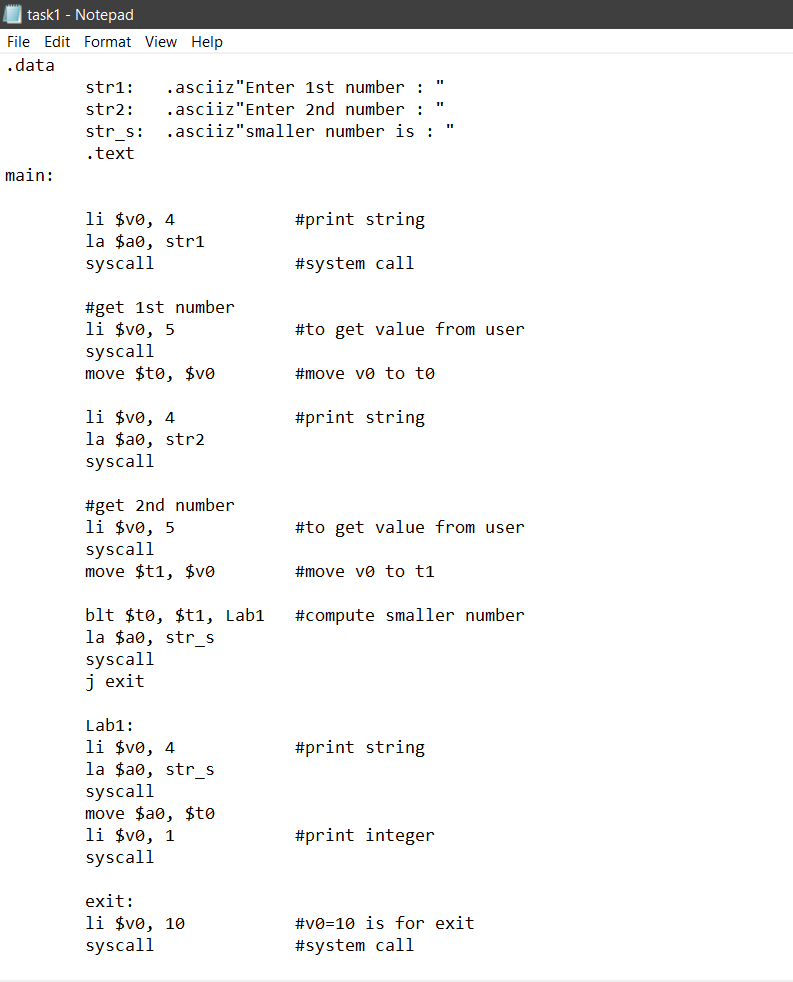
**University of Engineering and Technology, Peshawar**

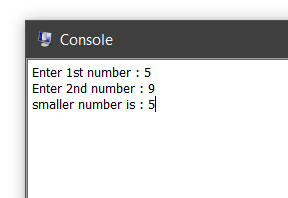
**Logical Instruction, Use of lo and hi Registers through, Multiplication and Division & Character Manipulation**

**Task # 01**

blt $t0, $t1, Lab1 # Branch if $t0 < $t1

**Code:**

** Output:**

****

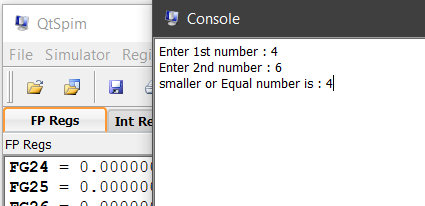
**Task # 02**

ble $t0, $t1, Lab2 # Branch if $t0 <= $t1

**Code:**

|  |
| --- |
|  |

**Output:**



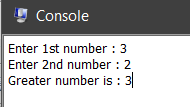
**Task # 03**

bgt $t0, $t1, Lab3 # Branch if $t0 > $t1

**Code:**

|  |
| --- |
|  |

**Output:**



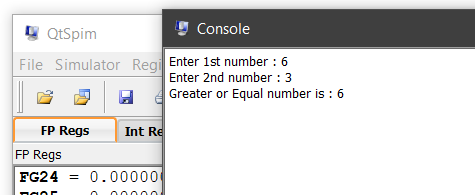
**Task # 04**

**bge $t0, $t1, Lab4 # Branch if $t0 >= $t1**

**Code:**

|  |
| --- |
|  |

**Output:**

****

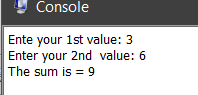
**Task # 5**

Add values.

**Code:**

|  |
| --- |
|  |

**Output:**

****

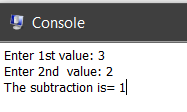
**Task # 6**

**Subtraction**

**Code:**

|  |
| --- |
| .data  str1: .asciiz"Enter 1st value: "  str2: .asciiz"Enter 2nd value: "  str3: .asciiz"subtraction = "  .text  main:    li $v0, 4 #print string  la $a0, str1  syscall #make sys call  #get the first no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t0, $v0 #move the no read into #t0  li $v0, 4 #print string 1  la $a0, str2  syscall #make sys call  #get the second no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t1, $v0 #move the no read into #t1    sub $t2, $t0, $t1 #compute the subtraction  li $v0, 4 #print string 2  la $a0, str3  syscall #make sys call  #print out #t2  move $a0, $t2  li $v0, 1 #load system call print\_int  syscall  li $v0, 10 #sys call code 10 is for exit  syscall #make sys call |

**Output:**

****

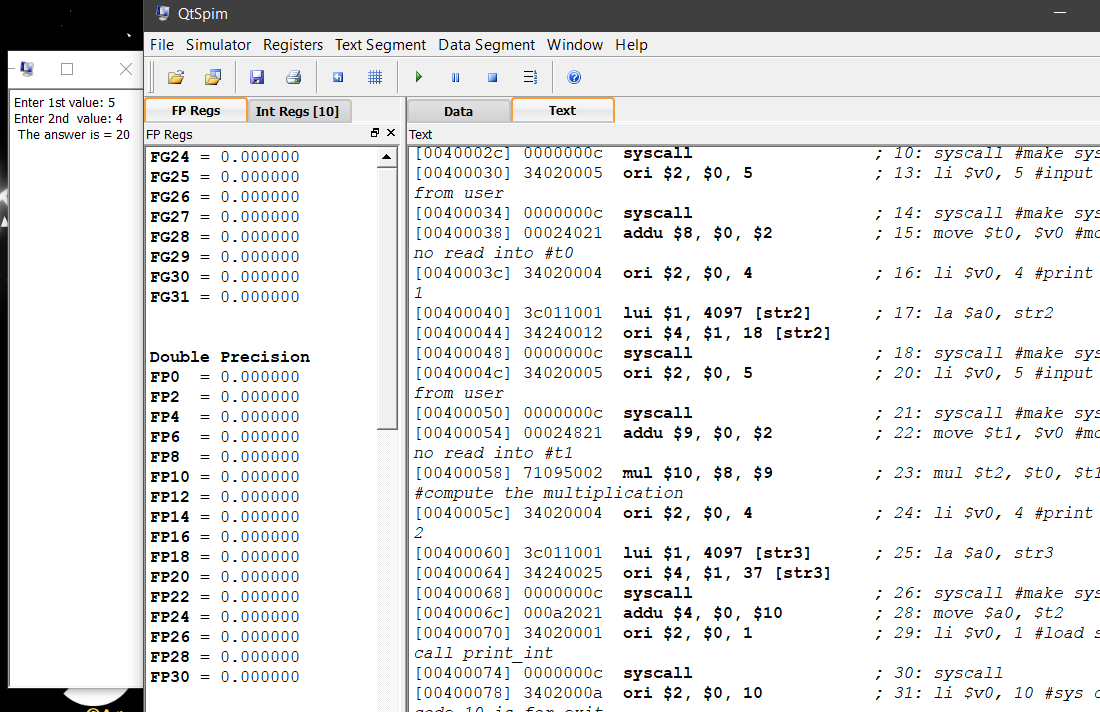
**Task # 7**

Multiplication

**Code:**

|  |
| --- |
| .data  str1: .asciiz"Enter 1st value: "  str2: .asciiz"Enter 2nd value: "  str3: .asciiz"Multiplication = "  .text  main:    li $v0, 4 #print string  la $a0, str1  syscall #make sys call  #get the first no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t0, $v0 #move the no read into #t0  li $v0, 4 #print string 1  la $a0, str2  syscall #make sys call  #get the second no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t1, $v0 #move the no read into #t1  mul $t2, $t0, $t1 #compute the multiplication  li $v0, 4 #print string 2  la $a0, str3  syscall #make sys call  #print out #t2  move $a0, $t2  li $v0, 1 #load system call print\_int  syscall  li $v0, 10 #sys call code 10 is for exit  syscall #make sys call |

**Output:**

****

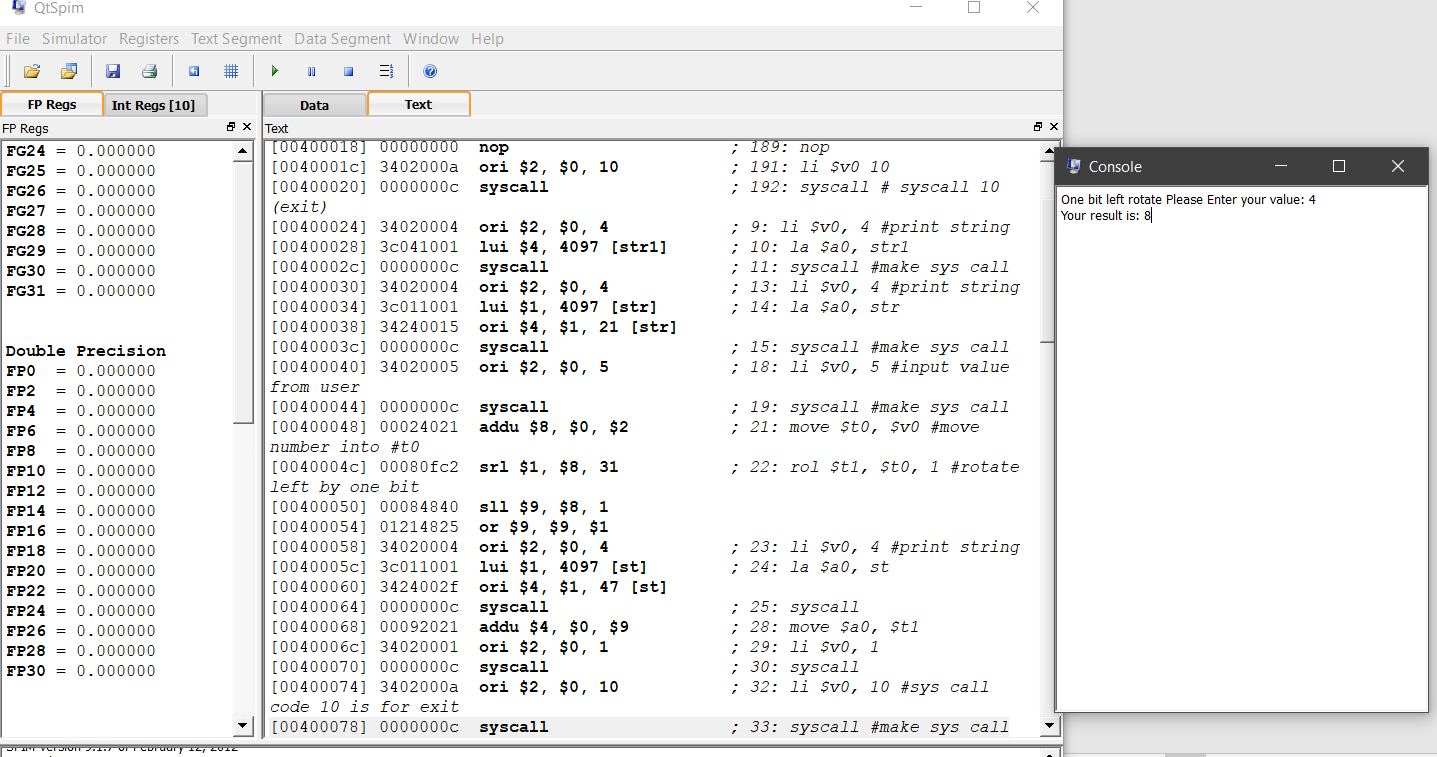
**Task # 8**

Rotate left by one bit

**Code:**

|  |
| --- |
| .data  str1: .asciiz"One bit left rotate "  str: .asciiz"Please Enter your value: "  st: .asciiz"Your result is: "  .text  main:  li $v0, 4 #print string  la $a0, str1  syscall #make sys call  li $v0, 4 #print string  la $a0, str  syscall #make sys call  #get the no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t0, $v0 #move number into #t0  rol $t1, $t0, 1 #rotate left by one bit  li $v0, 4 #print string  la $a0, st  syscall  #print out #t1  move $a0, $t1  li $v0, 1  syscall  li $v0, 10 #sys call code 10 is for exit  syscall #make sys call |

**Output:**



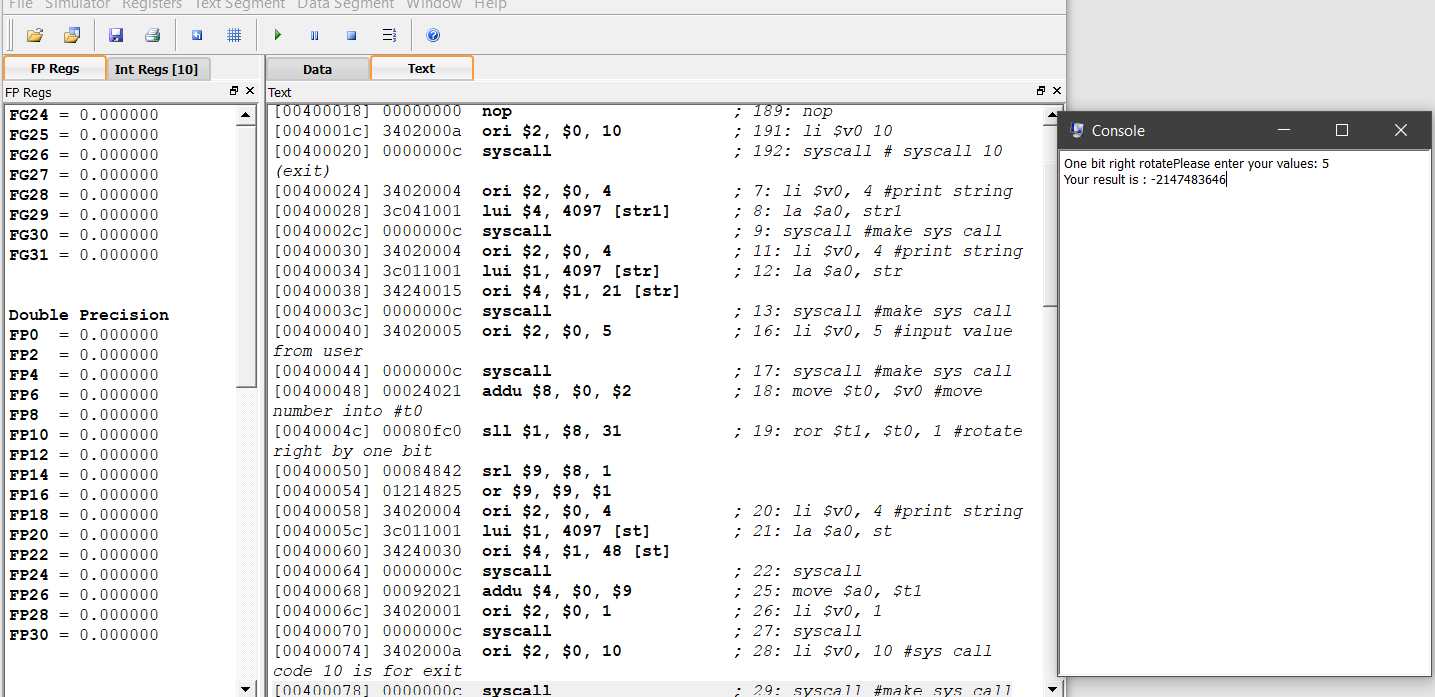
**Task # 9**

Rotate right by one bit

**Code:**

|  |
| --- |
| .data  str1: .asciiz"One bit right rotate"  str: .asciiz"Please enter your values: "  st: .asciiz"Your result is : "  .text  main:  li $v0, 4 #print string  la $a0, str1  syscall #make sys call  li $v0, 4 #print string  la $a0, str  syscall #make sys call  #get the no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t0, $v0 #move number into #t0  ror $t1, $t0, 1 #rotate right by one bit  li $v0, 4 #print string  la $a0, st  syscall  #print out #t1  move $a0, $t1  li $v0, 1  syscall  li $v0, 10 #sys call code 10 is for exit  syscall #make sys call |

**Output:**

****

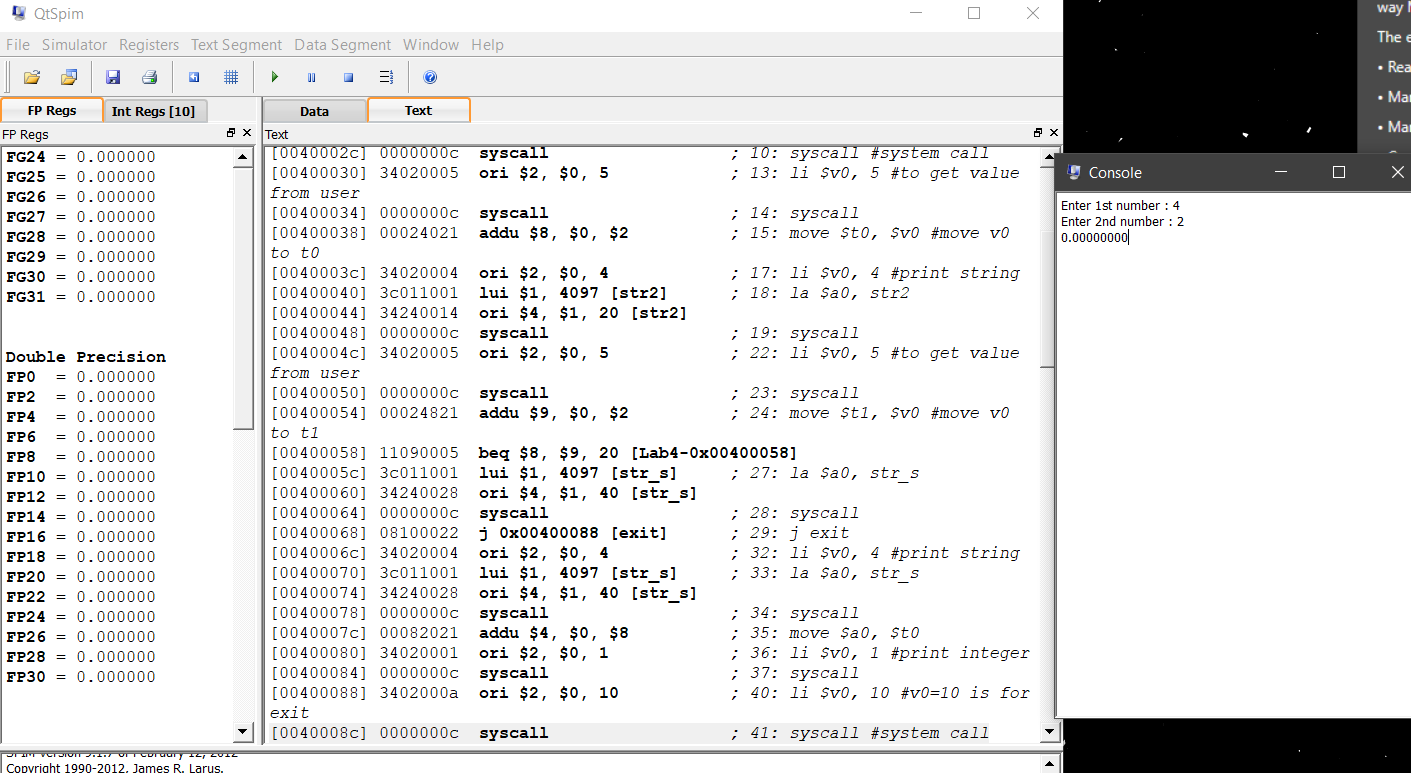
**Task # 10**

Binary equal(beq)

**Code:**

|  |
| --- |
| .data  str1: .asciiz"Enter 1st number : "  str2: .asciiz"Enter 2nd number : "  str\_s: .asciiz"Equal number is : "  .text  main:    li $v0, 4 #print string  la $a0, str1  syscall #system call  #get 1st number  li $v0, 5 #to get value from user  syscall  move $t0, $v0 #move v0 to t0    li $v0, 4 #print string  la $a0, str2  syscall  #get 2nd number  li $v0, 5 #to get value from user  syscall  move $t1, $v0 #move v0 to t1    beq $t0, $t1, Lab4 #binary equal  la $a0, str\_s  syscall  j exit    Lab4:  li $v0, 4 #print string  la $a0, str\_s  syscall  move $a0, $t0  li $v0, 1 #print integer  syscall    exit:  li $v0, 10 #v0=10 is for exit  syscall #system call |

**Output:**



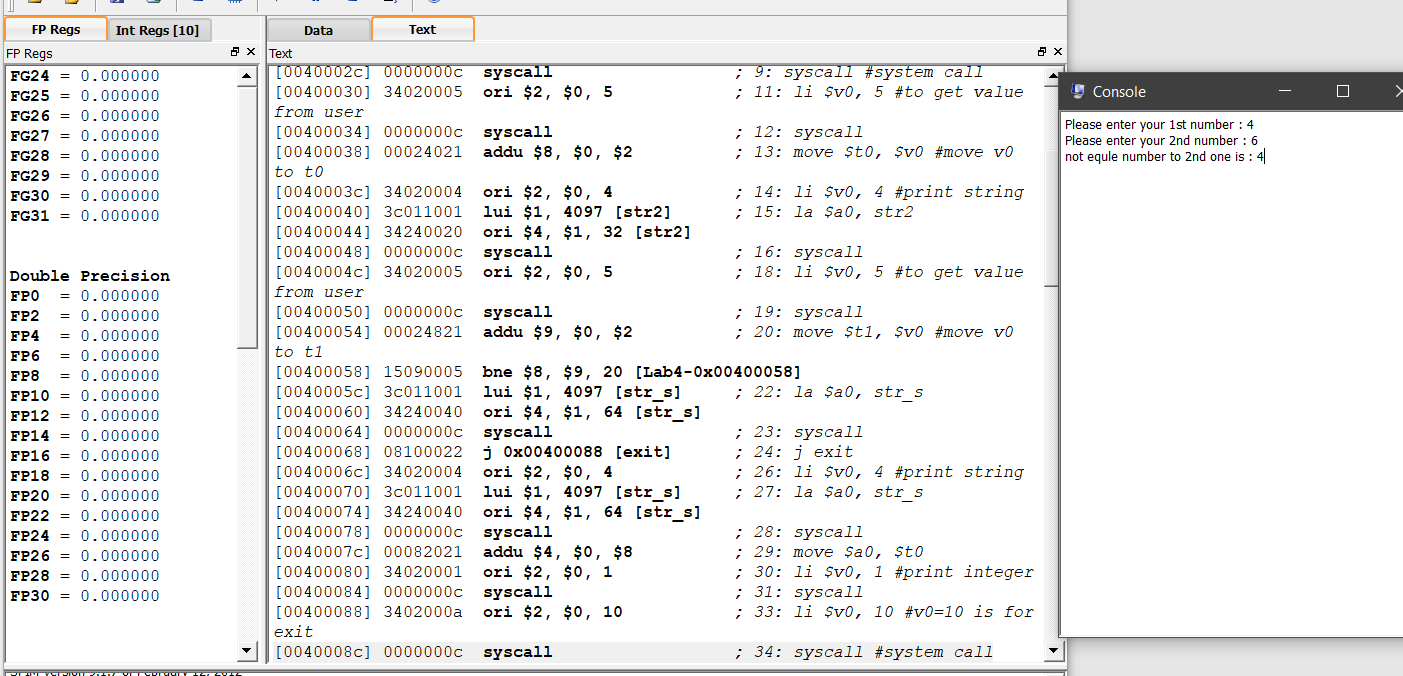
**Task # 11**

Binary Not equal (bne)

**Code:**

|  |
| --- |
|  |

**Output:**

****

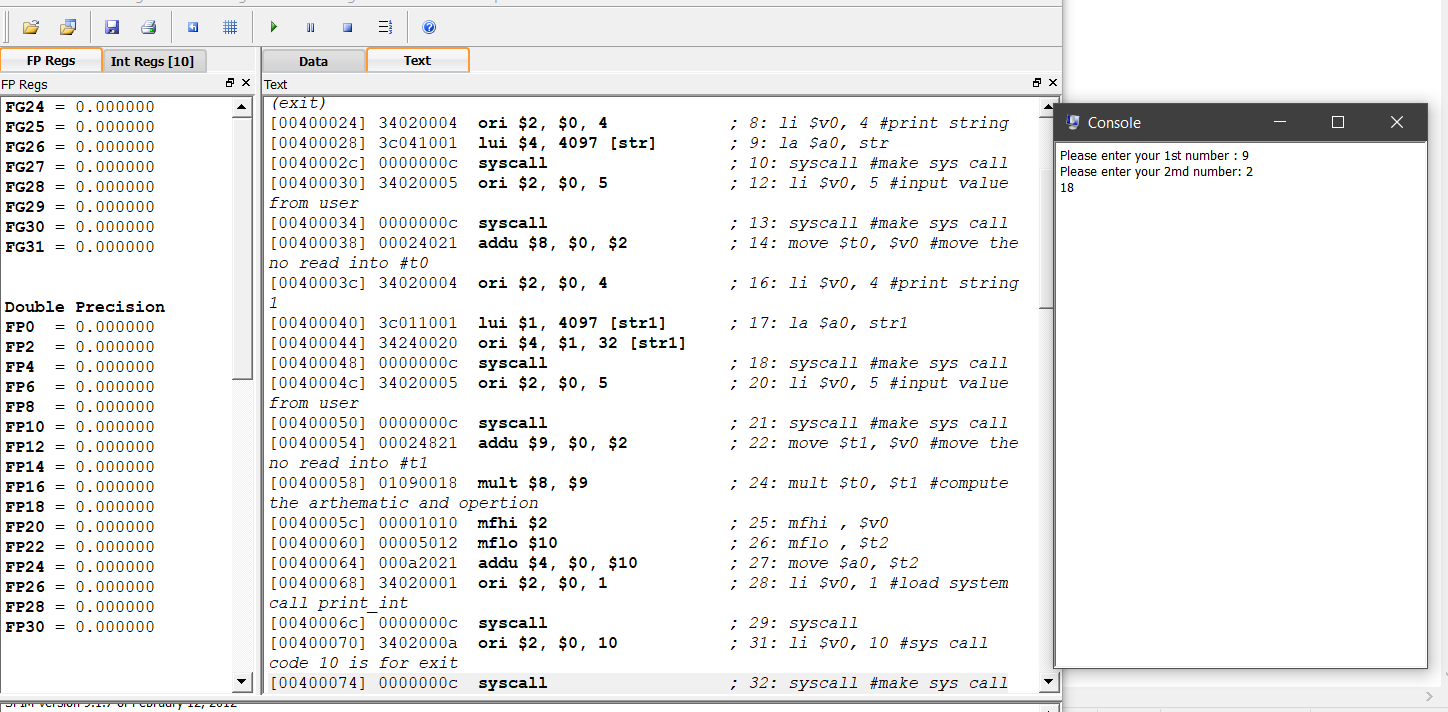
**Task # 12**

Mult

**Code:**

|  |
| --- |
| .data  str: .asciiz"Please enter your 1st number : "  str1: .asciiz"Please enter your 2md number: "  str2: .asciiz"Your Result is : "  .text  main:    li $v0, 4 #print string  la $a0, str  syscall #make sys call  #get the first no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t0, $v0 #move the no read into #t0    li $v0, 4 #print string 1  la $a0, str1  syscall #make sys call  #get the second no from the user  li $v0, 5 #input value from user  syscall #make sys call  move $t1, $v0 #move the no read into #t1    mult $t0, $t1 #compute the arthematic and opertion  mfhi , $v0  mflo , $t2  move $a0, $t2  li $v0, 1 #load system call print\_int  syscall  li $v0, 10 #sys call code 10 is for exit  syscall #make sys call |

**Output:**

****

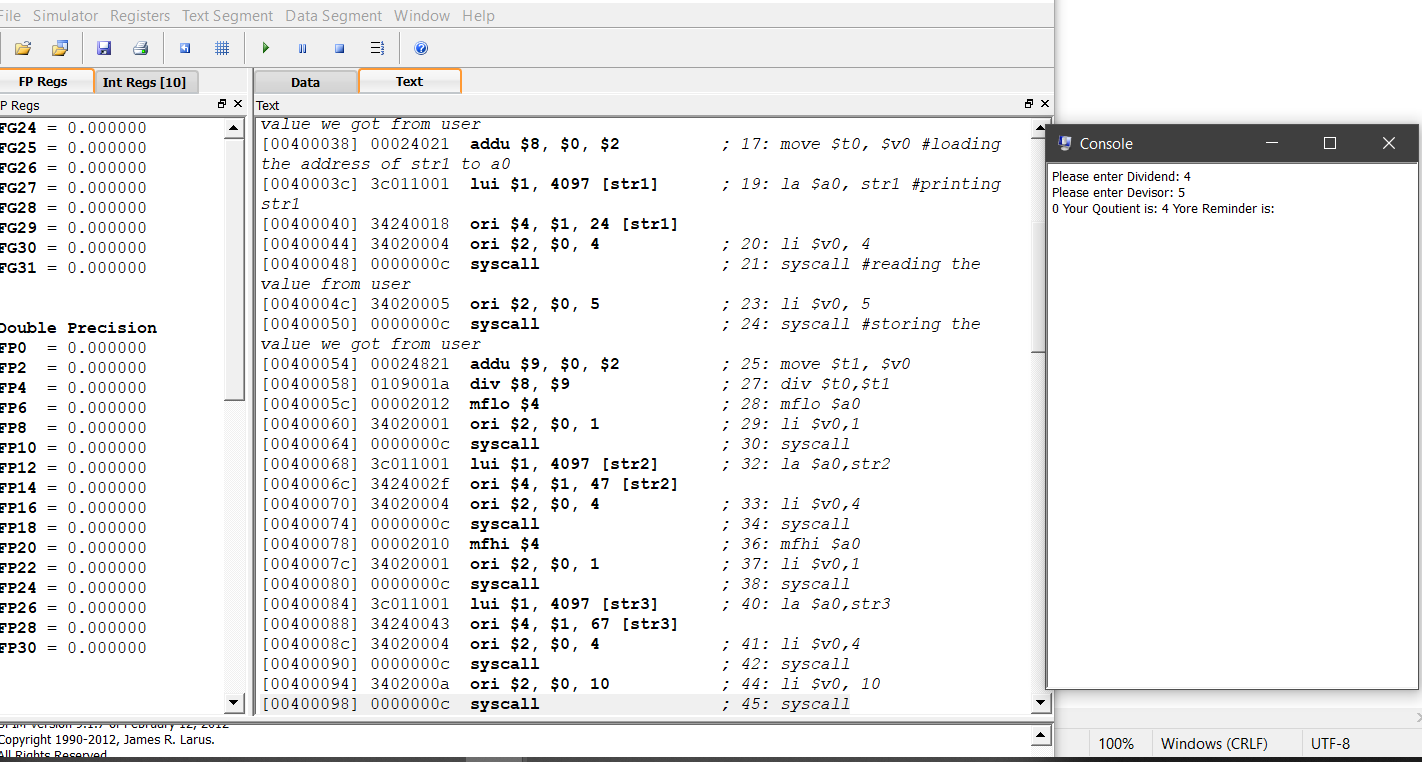
**Task # 13**

Divide(div)

**Code:**

|  |
| --- |
| .data  str: .asciiz "Please enter Dividend: "  str1: .asciiz "Please enter Devisor: "  str2: .asciiz " Your Qoutient is: "  str3: .asciiz " Yore Reminder is: "  .text  main:  #loading address of str to a0  la $a0, str #printing str  li $v0, 4  syscall #reading value from user    li $v0, 5  syscall #storing the value we got from user  move $t0, $v0 #loading the address of str1 to a0  la $a0, str1 #printing str1  li $v0, 4  syscall #reading the value from user    li $v0, 5  syscall #storing the value we got from user  move $t1, $v0  div $t0,$t1  mflo $a0  li $v0,1  syscall    la $a0,str2  li $v0,4  syscall  mfhi $a0  li $v0,1  syscall    la $a0,str3  li $v0,4  syscall  li $v0, 10  syscall |

**Output:**

****