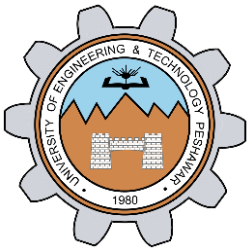
# COA Lab

**LAB # 02**




**Fall 2020**

**CSE304L Computer Organization and Architecture Lab**

Submitted by: **Shah Raza**

Registration No. : **18PWCSE1658**

Class Section: **B**

"On my honor, as student of University of Engineering and Technology, I have neither given nor received unauthorized assistance on this academic work."

Student Signature: _____

Submitted to:

**Engr. Ammad Khalil**

December 1, 2020

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

## Activity # 01

## Title:

Write the mips assembly code for the following logical instructions mentioned below. Take numbers from user and display the result on console. The lab report should contain code along with registers and console pic.

## Source code:

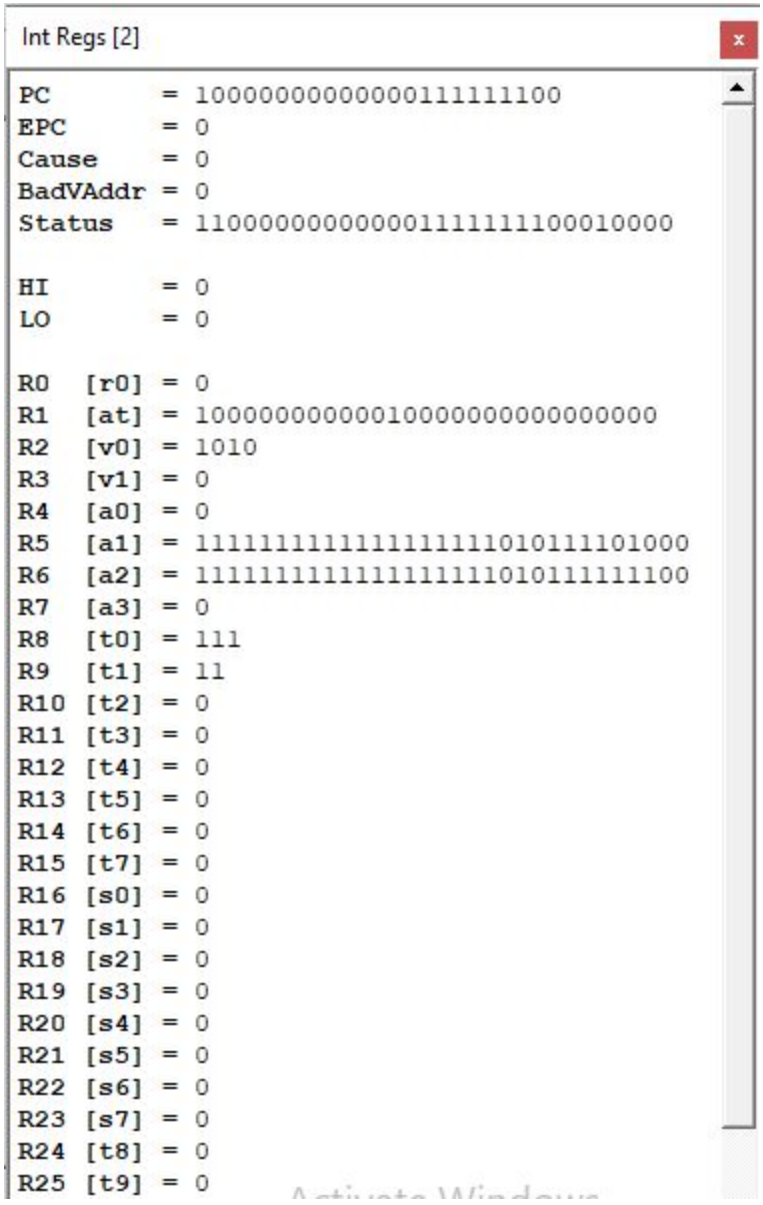
```
.data
str: .asciiz"Enter first number: "
str1: .asciiz"Enter second number: "
str2: .asciiz"And: "
str3: .asciiz"   Or: "
str4: .asciiz"   Xor: "
str5: .asciiz"   Nor: "
str6: .asciiz"   Andi: "
str7: .asciiz"   Ori: "
str8: .asciiz"   Xori: "
str9: .asciiz"   Shift left: "
str10: .asciiz"   Shift Right: "
str11: .asciiz"   Shift right arithm: "
str12: .asciiz"   Shift left by var: "
str13: .asciiz"   Shift right by var: "
str14: .asciiz"   Shift right arithm. by var: "
.text


main:
li $v0,4      #system call code for printing string
la $a0,str    #address of a string to print
syscall       #print the string
li $v0,5      #take the value from user
syscall
move $t0,$v0  #move the value to t0

li $v0,4      #system call code for printing string
la $a0,str1   #address of a string to print
syscall       #print the string
li $v0,5      #take the value from user
syscall
move $t1,$v0  #move the value to t1
```

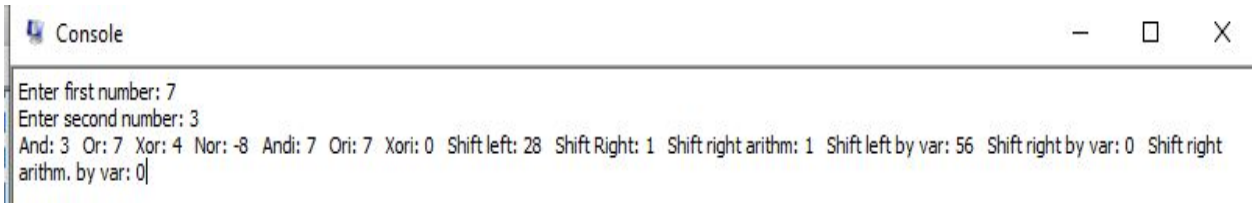
```
li $v0,4      #system call code for printing string
la $a0,str2   #address of a string to print
syscall       #print the string

and $t2,$t0,$t1 #and of t0 and t1 stored in t2
move $a0,$t2    #move t2 to a0
li $v0,1        #output the value
syscall

li $v0,4      #system call code for printing string
la $a0,str3   #address of a string to print
syscall       #print the string

or $t2,$t0,$t1  #or of t0 and t1 stored in t2
move $a0,$t2    #move t2 to a0
li $v0,1        #output the value
syscall

li $v0,4     #system call code for printing string
la $a0,str4  #address of a string to print
syscall      #print the string

xor $t2,$t0,$t1  #xor of t0 and t1 stored in t2
move $a0,$t2     #move t2 to a0
li $v0,1         #output the value
syscall

li $v0,4     #system call code for printing string
la $a0,str5  #address of a string to print
syscall      #print the string

nor $t2,$t0,$t1  #nor of t0 and t1 stored in t2
move $a0,$t2     #move t2 to a0
li $v0,1         #output the value
syscall

li $v0,4     #system call code for printing string
la $a0,str6  #address of a string to print
syscall      #print the string

andi $t2,$t0,7  #andi of t0 and 7 stored in t2
move $a0,$t2    #move t2 to a0
li $v0,1        #output the value
syscall

li $v0,4      #system call code for printing string
la $a0,str7   #address of a string to print
```

```
syscall      #print the string

ori $t2,$t0,7  #ori of t0 and 7 stored in t2
move $a0,$t2   #move t2 to a0
li $v0,1       #output the value
syscall

li $v0,4      #system call code for printing string
la $a0,str8   #address of a string to print
syscall       #print the string

xori $t2,$t0,7  #xori of t0 and 7 stored in t2
move $a0,$t2    #move t2 to a0
li $v0,1        #output the value
syscall

li $v0,4      #system call code for printing string
la $a0,str9   #address of a string to print
syscall       #print the string

sll $t2,$t0,2  #sll of t0 and 2 stored in t2
move $a0,$t2   #move t2 to a0
li $v0,1       #output the value
syscall

li $v0,4       #system call code for printing string
la $a0,str10   #address of a string to print
syscall        #print the string

srl $t2,$t0,2  #srl of t0 and 2 stored in t2
move $a0,$t2   #move t2 to a0
li $v0,1       #output the value
syscall

li $v0,4       #system call code for printing string
la $a0,str11   #address of a string to print
syscall        #print the string

sra $t2,$t0,2  #sra of t0 and 2 stored in t2
move $a0,$t2   #move t2 to a0
li $v0,1       #output the value
syscall


li $v0,4       #system call code for printing string
la $a0,str12   #address of a string to print
syscall        #print the string
```

```
sllv $t2,$t0,$t1  #sllv of t0 and t1 stored in t2
move $a0,$t2      #move t2 to a0
li $v0,1          #output the value
syscall

li $v0,4       #system call code for printing string
la $a0,str13   #address of a string to print
syscall        #print the string

srlv $t2,$t0,$t1  #srlv of t0 and t1 stored in t2
move $a0,$t2      #move t2 to a0
li $v0,1          #output the value
syscall

li $v0,4       #system call code for printing string
la $a0,str14   #address of a string to print
syscall        #print the string

srav $t2,$t0,$t1  #srav of t0 and t1 stored in t2
move $a0,$t2      #move t2 to a0
li $v0,1          #output the value
syscall
li $v0,10      #exit program
syscall
```

## Output:


Int Regs [2]
PC = 10000000000000111111100
EPC = 0
Cause = 0
BadVAddr = 0
Status = 11000000000000111111111000010000
HI = 0
LO = 0
R0 [r0] = 0
R1 [at] = 10000000000010000000000000000
R2 [v0] = 1010
R3 [v1] = 0
R4 [a0] = 0
R5 [a1] = 11111111111111111111010111101000
R6 [a2] = 11111111111111111111010111111100
R7 [a3] = 0
R8 [t0] = 111
R9 [t1] = 11
R10 [t2] = 0
R11 [t3] = 0
R12 [t4] = 0
R13 [t5] = 0
R14 [t6] = 0
R15 [t7] = 0
R16 [s0] = 0
R17 [s1] = 0
R18 [s2] = 0
R19 [s3] = 0
R20 [s4] = 0
R21 [s5] = 0
R22 [s6] = 0
R23 [s7] = 0
R24 [t8] = 0
R25 [t9] = 0


Console
Enter first number: 7
Enter second number: 3
And: 3 Or: 7 Xor: 4 Nor: -8 Andi: 7 Ori: 7 Xori: 0 Shift left: 28 Shift Right: 1 Shift right arithm: 1 Shift left by var: 56 Shift right by var: 0 Shift right arithm. by var: 0