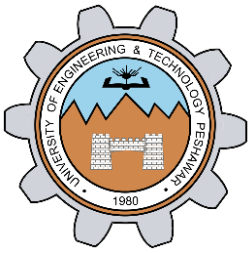
COA Lab

LAB #07



Fall 2020

CSE304L Computer Organization and Architecture Lab

Submitted by: **Shah Raza**

Registration No.: **18PWCSE1658**

Class Section: **B**

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

Student Signature: _____

Submitted to:

Engr. Amaad Khalil

16th February, 2021

Department of Computer Systems Engineering

University of Engineering and Technology, Peshawar

Task:

Write a mips assembly program using double precision floating point having 10 operations or instructions

Source code:

```
.data
	str: .asciiz"Enter first floating point number: "
	str1: .asciiz"Enter second floating point number: "
	str2: .asciiz"\nAdd: "
	str3: .asciiz"\nSub: "
	str4: .asciiz"\nDiv: "
	str5: .asciiz"\nMul: "
	str6: .asciiz"\nAbs of 1st number: "
	str7: .asciiz"\nNeg of 2nd number: "
	str8: .asciiz"\n1st number in int form: "
	str9: .asciiz"\n1st number is less than 2nd number."
	str10: .asciiz"\n1st number is less than or equal to 2nd number."
	str11: .asciiz"\n1st number is greater than 2nd number."
	str12: .asciiz"\n1st number is greater than or equal to 2nd number."
	str13: .asciiz"\n1st number equal to 2nd number."
	str14: .asciiz"\n1st number is not equal to 2nd number."
.text


main:

	li.d $f1,0.0  #f1=0.0

	li $v0,4      #system call code for printing string
	la $a0,str    #address of a string to print
	syscall       #print the string
	li $v0,7      #take the value from user
	syscall

	mov.d $f2,$f0
```

```
li $v0,4      #system call code for printing string
la $a0,str1   #address of a string to print
syscall       #print the string
li $v0,7      #take the value from user
syscall
mov.d $f4,$f0


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

add.d $f6,$f2,$f4

mov.d $f12,$f6
li $v0,3        #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

sub.d $f6,$f2,$f4

mov.d $f12,$f6
li $v0,3        #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

div.d $f12,$f2,$f4
```

```
li $v0,3          #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

mul.d $f12,$f2,$f4

li $v0,3          #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

abs.d $f12,$f2

li $v0,3          #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

neg.d $f12,$f4

li $v0,3          #output the value
syscall

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

```
	cvt.w.d $f12,$f2

	li $v0,3          #output the value
	syscall

	c.eq.d $f2,$f4

	bc1t Equal

	c.lt.d $f2,$f4

	bc1t LessThan

	c.le.d $f2,$f4

	bc1t LessThanOrEqual

	bc1f GreaterThan

	cvt.d.w $f5,$f2


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

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

GreaterThan:
```

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

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

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

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

exit:
	li $v0,10	#Terminate the program
	syscall			#syscall
```

Output:

```
Enter first double floating point number: 7.12
Enter second double floating point number: 3.54

Add: 10.66
Sub: 3.5800000000000001
Div: 2.0112994350282487
Mul: 25.204800000000002
Abs of 1st number: 7.1200000000000001
Neg of 2nd number: -3.54
Single floating point: -3.5399994916468858
Double floating point: 89128.9609375
```