**Bahria University, Lahore Campus**

Department of Computer Sciences

Lab Journal 03

**(Spring 2023)**

|  |  |  |
| --- | --- | --- |
| Course: | **Computer Architecture & Organization Lab** |  |
| Course Code: | CEL 221 | Max Marks: 30 |
| Faculty’s Name: | Maryam Munawar | Lab Engineer: |

Name: \_AFFAN AHMAD\_\_\_\_ Enroll No: \_03-134221-003\_\_\_

Ttask no 1

.data

a: .asciiz "Enter the first number :"

b: .asciiz "Enter the second number :"

c: .asciiz "your result is :"

.text

la $a0,a

li $v0,4

syscall

li $v0,5

syscall

move $t0,$v0

la $a0,b

li $v0,4

syscall

li $v0,5

syscall

move $t1 ,$v0

add $t2,$t0,$t1

la $a0,c

li $v0,4

syscall

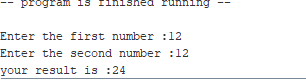
li $v0 ,1

add $a0,$zero,$t2

syscall

li $v0,10

syscall



## Lab Tasks:

### Task1: 10 Minutes

Purpose: To illustrate how to translate a pseudo code program into assembly

Pseudo Code

main

{

register int i = input("Please enter the first value to add: ");

register int j = input("Please enter the second value to add: ");

register int k = i + j;

print("The result is " + k);

}

### Task2: 10 Minutes

Purpose: To illustrate how to translate a pseudo code program into assembly

Pseudo Code

main

{

.data

o: .asciiz "Enter your number: "

result: .asciiz "A result of 0 is even, 1 is odd: result = "

.text

la $a0, o

li $v0, 4

syscall

li $v0,5

syscall

move $s0,$v0

addi $t0, $zero, 2

div $t0, $s0, $t0

mfhi $t1

li $v0, 4

la $a0, result

syscall

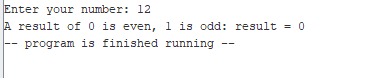
li $v0, 1

move $a0, $t1

syscall

li $v0,10

syscall



int i = prompt("Enter your number");

int j = i % 2;

print("A result of 0 is even, a result of 1 is odd: result = " + j;

}

Purpose: To have a user enter a number,and print 0 if

the number is even, 1 if the number is odd

### Task3: 10 Minutes

Purpose: To illustrate how to translate a pseudo code program into assembly

Pseudo Code

main

{

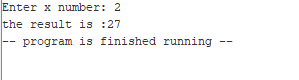
int x = prompt("Enter a value for x: ");

int y = 5 \* x \* x + 2 \* x + 3;

print("The result is: " + y);

}

Purpose: To calculate the result of 5\*x^ + 2\*x + 3



.data

o: .asciiz "Enter x number: "

result: .asciiz "the result is :"

.text

la $a0, o

li $v0, 4

syscall

li $v0,5

syscall

move $s0,$v0

mul $t0, $s0, $s0

mul $t0, $t0, 5

mul $t1, $s0, 2

add $t0, $t0, $t1

add $t2, $t0, 3

li $v0, 4

la $a0, result

syscall

li $v0, 1

move $a0, $t2

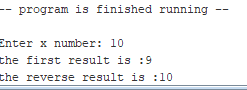
syscall

li $v0,10

syscall

### Task4: 10 Minutes

One thing to always keep in mind when using division with integers in any language (including Java, C/C++, etc) is that the results are truncated. This can lead to errors and different answers depending on the order of evaluation of the terms in the equation. For example, most 5th graders knows that "(10/3) \*3 = 10", as the 3's should cancel. However in integer arithmetic the result of "10/3 = 3", and so "(10/3) \*3 = 9" (not 10). However if you reverse the order of the operations you will find that "(10\*3) / 3 = 10". This is shown in the following program. Ordering of multiplication and division are reversed.



.data

o: .asciiz "Enter x number: "

result: .asciiz "the first result is :"

result2:.asciiz"\nthe reverse result is :"

.text

la $a0, o

li $v0, 4

syscall

li $v0,5

syscall

move $s0,$v0

div $t0, $s0, 3

mul $t1,$t0,3

li $v0, 4

la $a0, result

syscall

li $v0, 1

move $a0, $t1

syscall

mul $t1,$s0,3

div $t0, $t1, 3

li $v0, 4

la $a0, result2

syscall

li $v0, 1

move $a0, $t0

syscall

li $v0,10

syscall

**Lab Grading Sheet :**

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Max Marks** | **Obtained Marks** | **Comments(*if any*)** |
| 1. | 10 |  |  |
| 2. | 10 |  |  |
| 3. | 10 |  |  |
| 4. | 10 |  |  |
| **Total** | **40** |  | **Signature** |

**Note : Attempt all tasks and get them checked by your Instructor**