



09/04/2024

LEAF EXAMPLE

Prepared For :
Dr.lamiaa

Description of the Problem:

Problem Statement:

Write a MIPS assembly language program that prompts the user to enter four integer values. After receiving the inputs, the program should pass these values to a subroutine named `leaf_example`. The `leaf_example` subroutine should perform arithmetic operations on the inputs and display the result.

SAMPLE INPUT:

```
ENTER THE VALUE: 10
ENTER THE VALUE: 5
ENTER THE VALUE: 3
ENTER THE VALUE: 7
```

SAMPLE OUTPUT:

```
THE OUTPUT IS: 5
```

Updates Added to Make the Example Work:

1-Initialization:

The program initializes the message strings for input and output using `.asciiz`.

The `.data` section is added to define these strings.

The `.text` section is added to start the code segment.

2-Prompting for Input:

The program uses `syscall 4` to prompt the user to enter a value.

It uses `syscall 5` to read an integer input from the user and stores it in registers `$t0`, `$t1`, `$t2`, and `$t3`.

Passing Values to `leaf_example`:

The program moves the input values into arguments `$a0`, `$a1`, `$a2`, and `$a3`, which are used to pass arguments to a function.

The `jal` instruction is used to jump and link to the `leaf_example` subroutine.:

3-Passing Values to leaf_example:

The program moves the input values into arguments \$a0, \$a1, \$a2, and \$a3, which are used to pass arguments to a function.

The jal instruction is used to jump and link to the leaf_example subroutine.

4-leaf_example Subroutine:

The leaf_example subroutine performs arithmetic operations on the input values.

It adds the first two inputs (\$a0 and \$a1) and stores the result in \$t0.

It adds the next two inputs (\$a2 and \$a3) and stores the result in \$t1.

It subtracts the result of the second operation from the first (\$t0 - \$t1) and stores the result in \$s0.

Displaying Output:

The program uses syscall 1 to display the output result stored in \$s0.

Cleaning Up and Exiting:

The program restores the stack pointer and return address before exiting the leaf_example subroutine.

It exits the program using syscall 10.

this our screen shot for input and output and there is a register in picture \$t0,\$t1,\$t2,\$t3 then this moved to \$a0,\$a1,\$a2,and \$a3 then print the value

C:\Users\MF\Desktop\assembly\leaf_exmple.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Text Segment

Byte	Address	Code	Basic	Source
	4194304	0x24020004	addiu \$2,\$0,4	8: li \$v0, 4
	4194308	0x3c011001	lui \$1,4097	9: la \$a0, input
	4194312	0x34240000	ori \$4,\$1,0	
	4194316	0x0000000c	syscall	10: syscall
	4194320	0x24020005	addiu \$2,\$0,5	13: li \$v0, 5
	4194324	0x0000000c	syscall	14: syscall
	4194328	0x00024021	addiu \$8,\$0,\$2	15: move \$t0, \$v0
	4194332	0x24020004	addiu \$2,\$0,4	18: li \$v0, 4
	4194336	0x3c011001	lui \$1,4097	19: la \$a0, input
	4194340	0x34240000	ori \$4,\$1,0	
	4194344	0x0000000c	syscall	20: syscall
	4194348	0x24020005	addiu \$2,\$0,5	23: li \$v0, 5
	4194352	0x0000000c	syscall	24: syscall
	4194356	0x00024021	addiu \$9,\$0,\$2	25: move \$t1, \$v0
	4194360	0x24020004	addiu \$2,\$0,4	28: li \$v0, 4
	4194364	0x3c011001	lui \$1,4097	29: la \$a0, input
	4194368	0x34240000	ori \$4,\$1,0	

Labels

Label	Address
(global)	
main	4194304
leaf_exmple.asm	
leaf_example	4194460
input	268500992
output	268501010

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+12)	Value (+16)	Value (+20)	Value (+24)	Value (+28)
268500992	1702129221	1752440946	1635131493	979727724	1750335520	1970217061	1953853556	980642080
268501024	32	0	0	0	0	0	0	0
268501056	0	0	0	0	0	0	0	0
268501088	0	0	0	0	0	0	0	0
268501120	0	0	0	0	0	0	0	0
268501152	0	0	0	0	0	0	0	0
268501184	0	0	0	0	0	0	0	0
268501216	0	0	0	0	0	0	0	0
268501248	0	0	0	0	0	0	0	0
268501280	0	0	0	0	0	0	0	0
268501312	0	0	0	0	0	0	0	0
268501344	0	0	0	0	0	0	0	0
268501376	0	0	0	0	0	0	0	0
268501408	0	0	0	0	0	0	0	0

Registers

Name	Number	Value
\$zero	0	0
\$at	1	268500992
\$v0	2	10
\$v1	3	0
\$a0	4	5
\$a1	5	5
\$a2	6	3
\$a3	7	7
\$t0	8	15
\$t1	9	10
\$t2	10	3
\$t3	11	7
\$t4	12	0
\$t5	13	0
\$t6	14	0
\$t7	15	0
\$s0	16	0
\$s1	17	0
\$s2	18	0
\$s3	19	0
\$s4	20	0
\$s5	21	0
\$s6	22	0
\$s7	23	0
\$s8	24	0
\$s9	25	0
\$k0	26	0
\$k1	27	0
\$gp	28	268468224
\$sp	29	2147479548
\$fp	30	0
\$ra	31	4194460
\$PC		4194460
\$hi		0
\$lo		0

Mars Messages Run IO

Enter the value: 10
Enter the value: 5
Enter the value: 3
Enter the value: 7
The output is: 5
-- program is finished running --

Clear