



University of Colombo School of Computing

SCS 3211 : Compiler Theory
Activity 02

Pasindu Fernando : 20000512

```

1  #20000512 - Pasindu Fernando
2  .data
3      Compiler_theory_credit:    .float 2.0
4      Graph_theory_credit:      .float 1.0
5      Programming_credit:       .float 3.0
6      A_GPV:                   .float 4.0
7      B_GPV:                   .float 3.3
8      C_GPV:                   .float 2.0
9      final_gpa:               .float 0.0
10
11 .text
12 .globl main
13 main:
14     # Calculate the GPA
15     jal    calculate_gpa
16
17     # Print the final GPA
18     li     $v0, 2                # Set $v0 to 2 for printing a float value
19     lwcl   $f12, final_gpa      # Load final GPA into $f12
20     syscall
21
22     # Terminate the program
23     li     $v0, 10              # Set $v0 to 10 for program exit
24     syscall
25
26 calculate_gpa:
27     # Load credit values
28     lwcl   $f0, Compiler_theory_credit # Load Compiler_theory_credit into $f0
29     lwcl   $f1, Graph_theory_credit   # Load Graph_theory_credit into $f1
30     lwcl   $f2, Programming_credit    # Load Programming_credit into $f2
31
32     # Calculate total credit value
33     add.s   $f3, $f0, $f1            # Add Compiler_theory_credit and Graph_theory_credit
34     add.s   $f3, $f3, $f2            # Add $f3 and Programming_credit
35
36     # Load grade point values
37     lwcl   $f4, A_GPV                # Load A_GPV into $f4
38     lwcl   $f5, B_GPV                # Load B_GPV into $f5
39     lwcl   $f6, C_GPV                # Load C_GPV into $f6
40
41     # Calculate GPA
42     mul.s   $f7, $f0, $f4            # Multiply Compiler_theory_credit by A_GPV
43     mul.s   $f8, $f1, $f5            # Multiply Graph_theory_credit by B_GPV
44     mul.s   $f9, $f2, $f6            # Multiply Programming_credit by C_GPV
45
46     add.s   $f10, $f7, $f8           # Add $f7 (Compiler_theory_credit * A_GPV) and $f8 (Graph_theory_credit * B_GPV)
47     add.s   $f10, $f10, $f9          # Add $f10 and $f9 (Programming_credit * C_GPV)
48
49     div.s   $f11, $f10, $f3          # Divide $f10 (GPA sum) by $f3 (total credit sum)
50
51     mov.s   $f12, $f11              # Move final GPA to $f12
52     swcl   $f12, final_gpa          # Store final GPA in memory
53
54     jr     $ra
55

```

Result

C:\Users\Pasindu\Desktop\Third Year\Compiler theory\MIPS5Activity2.asm - MARS 4.5

File Edit Run Settings Tools Help

Run speed at max (no interaction)

Edit Execute

Text Segment

Bkpt	Address	Code	Basic	Source
	0x00400000	0x00100007	jal 0x0040001c	15: jal calculate_gpa
	0x00400004	0x24020002	addiu \$2,\$0,0x00000018	18: li \$v0, 2 # Set \$v0 to 2 for printing a float value
	0x00400008	0x3c011001	lwl \$f12,0x000001001	19: lwl \$f12, final_gpa # Load final GPA into \$f12
	0x0040000c	0xc42e0018	lwl \$f12,0x0000001...	
	0x00400010	0x0000000c	syscall	20: syscall
	0x00400014	0x2402000a	addiu \$2,\$0,0x00000023	23: li \$v0, 10 # Set \$v0 to 10 for program exit

Data Segment

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x10010020	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010040	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010060	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010080	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100100e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010100	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010120	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010140	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010160	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x10010180	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101a0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101c0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000
0x100101e0	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000	0x00000000

0x10010000 (.data) [x] Hexadecimal Addresses [x] Hexadecimal Values [] ASCII

Mars Messages Run I/O

```
2.8833332
-- program is finished running --
```

Clear

Registers Coproc 1 Coproc 0

Name	Number	Value
\$zero	0	0x00000000
\$at	1	0x10010000
\$v0	2	0x0000000a
\$v1	3	0x00000000
\$a0	4	0x00000000
\$a1	5	0x00000000
\$a2	6	0x00000000
\$a3	7	0x00000000
\$t0	8	0x00000000
\$t1	9	0x00000000
\$t2	10	0x00000000
\$t3	11	0x00000000
\$t4	12	0x00000000
\$t5	13	0x00000000
\$t6	14	0x00000000
\$t7	15	0x00000000
\$n0	16	0x00000000
\$n1	17	0x00000000
\$n2	18	0x00000000
\$n3	19	0x00000000
\$n4	20	0x00000000
\$n5	21	0x00000000
\$n6	22	0x00000000
\$n7	23	0x00000000
\$n8	24	0x00000000
\$n9	25	0x00000000
\$k0	26	0x00000000
\$k1	27	0x00000000
\$gp	28	0x10008000
\$sp	29	0x7ffffeffc
\$fp	30	0x00000000
\$ra	31	0x0040001c
pc		0x0040001c
hi		0x00000000
lo		0x00000000