CLEVELAND STATE UNIVERSITY

Class: *CIS580 (Computer Architecture)*
Project Title: *MIPS programming (project#01)*
Name: *Daniel Izadnegahdar / 2420596*
Submission Login ID#: *daizadne*

1. *Summary of code:*

   1.1. This code was developed with MIPS, using the MARS simulator IDE developed by *Missouri State University.* The program uses assembly to input 2 integers to perform a sum, subtraction, multiplication, and division.

      1.1.1. *Program summary:*

         1.1.1.1. *Data block:* The program begins by defining the intro, prompt, and output strings. This block is defined using the .asciiz assembly call.
         1.1.1.2. *Print intro:* Next, the program prints the intro to the program. It does this by using the li command, where 4 is used to print. Next, it uses la to define the string to print. And finally, syscall is used to execute the command.
         1.1.1.3. *Receive inputs:* Next, the program displays the prompts to properly receive integers from the user. It reads user input by using the li command, syscall, and the move command. The inputs are stored inside $t1 and $t0.
         1.1.1.4. *Calculate sum:* Next, the sum is calculated by adding $t1 and $t0 by using the add function. The numbers are printed per item#1.1.1.2.
         1.1.1.5. *Calculate difference:* Next, the difference is calculated by subtracting $t1 and $t0 by using the sub function. The numbers are then printed per item#1.1.1.2.
         1.1.1.6. *Calculate multiplication:* Next, the product is calculated by multiplying $t1 and $t0 by using the mul function. The numbers are then printed per item#1.1.1.2.
         1.1.1.7. *Calculate division:* Next, the result is calculated by dividing $t1 and $t0 by using the div function. The remainder is calculated using the rem function. The result is printed per item#1.1.1.2.

2. *Run instructions:*

   2.1. *Keyboard method:*
      2.1.1. Run the MARS program and open the provided cis580_lab01.asm file
      2.1.2. *Keyboard:* Press F3 to assemble and F5 to run.
      2.1.3. *GUI (Alternative):* Press F3 to assemble and F5 to run.

3. *Sample test run:*

```
Mars Messages    Run I/O

         Name: Daniel Izadnegahdar
         Title: CIS580 Project#01
         Description: MIPS program that performs basic arithmetic operations on 2 numbers:

         Enter 2 numbers and I'll show you the sum,
         Difference, product, quotient, and remainder.

         1st number: 9
         2nd number: 2

Clear    9 + 2 = 11
         9 - 2 = 7
         9 X 2 = 18
         9 / 2 = 4 R 1

         These are your results, Goodbye!
         -- program is finished running --
```

4. Registers:

Registers	Coproc 1	Coproc 0
Name	Number	Value
$zero	0	0x00000000
$at	1	0x10010000
$v0	2	0x0000000a
$v1	3	0x00000000
$a0	4	0x1001011b
$a1	5	0x00000000
$a2	6	0x00000000
$a3	7	0x00000000
$t0	8	0x00000009
$t1	9	0x00000002
$t2	10	0x00000000
$t3	11	0x00000000
$t4	12	0x00000000
$t5	13	0x00000000
$t6	14	0x00000000
$t7	15	0x00000000
$s0	16	0x00000000
$s1	17	0x00000000
$s2	18	0x00000000
$s3	19	0x00000000
$s4	20	0x00000000
$s5	21	0x00000000
$s6	22	0x00000000
$s7	23	0x00000000
$t8	24	0x00000000
$t9	25	0x00000000
$k0	26	0x00000000
$k1	27	0x00000000
$gp	28	0x10008000
$sp	29	0x7fffeffc
$fp	30	0x00000000
$ra	31	0x00000000
pc		0x00400224
hi		0x00000001
lo		0x00000004

Registers	Coproc 1	Coproc 0
Name	Float	Double
$f0	0x00000000	0x0000000000000000
$f1	0x00000000	
$f2	0x00000000	0x0000000000000000
$f3	0x00000000	
$f4	0x00000000	0x0000000000000000
$f5	0x00000000	
$f6	0x00000000	0x0000000000000000
$f7	0x00000000	
$f8	0x00000000	0x0000000000000000
$f9	0x00000000	
$f10	0x00000000	0x0000000000000000
$f11	0x00000000	
$f12	0x00000000	0x0000000000000000
$f13	0x00000000	
$f14	0x00000000	0x0000000000000000
$f15	0x00000000	
$f16	0x00000000	0x0000000000000000
$f17	0x00000000	
$f18	0x00000000	0x0000000000000000
$f19	0x00000000	
$f20	0x00000000	0x0000000000000000
$f21	0x00000000	
$f22	0x00000000	0x0000000000000000
$f23	0x00000000	
$f24	0x00000000	0x0000000000000000
$f25	0x00000000	
$f26	0x00000000	0x0000000000000000
$f27	0x00000000	
$f28	0x00000000	0x0000000000000000
$f29	0x00000000	
$f30	0x00000000	0x0000000000000000
$f31	0x00000000	

5. *Experiences in debugging and testing:*

5.1. *Development:*

    5.1.1. The development began with research on *MIPS programming* using the *MARS IDE*. This required review of documentation and the provided class material.

5.2. *Debugging & Testing:*

    5.2.1. To debug, the code errors were diagnosed by reviewing the output notifications. The run, step, and stop tools were used for troubleshooting.