CSCE 212 Project 3 Report

Tyler Beetle

1.0 Program Input and Output

For the first program we created, we first outputted a prompt to the user asking for 3 input values. We then took in 3 inputs from the user in the form of variables a,b, and c. Then, based off of the program, we returned a statement to let the user know which input had the greatest value. For the second program, We took in no input from the user. The necessary values came from the interior of the program. However, based on the unique values, we outputted the calculated variable BMI to the user. Based on this value we also returned if the value was in the underweight, overweight, or normal region. For the third and final program, we took in a total of four inputs from the user. We took in an homework count and an exercise count, along with an average hour value for each count. We then used this data to calculate and output the total number of hours the user spent on work.

2.0 Program Design

In order to start the first program, we initialized prompts to print out the user to start the program and to finish it. We then printed a prompt out asking the user for three inputs of numbers. We took each of these inputs and stored them in temporary registers. We then took each input and compared it to another in order to find which one had the greatest value. Once we found whichever one it was, we returned it back out to the user. For the second program, we started the program by creating 6 floating point constant numbers and 3 prompts for the end of the program. We then stored all of the fp numbers into registers. We then used the given values to calculate the BMI using the formula provided. After we found the BMI, we checked to see where it fell on the scale. If it was less than 18.5, the user was informed they were underweight and the value. If it was in between 18.5 and 24.9, the user was informed they were healthy. If it

was over 25, the user was informed they were overweight. All of this was done by comparing the calculated value to the interval values. The final program we created 4 prompts for the user. We then prompted the user to enter in their average homework time, amount of homework, average exercise time, amount of exercises. We stored all four of these values into registers. We then multiplied the hours times the amount for both the homework and the exercise. We then took these calculated values and added them together. We took this value and printed it back out to the user.

3.0 Symbol Table

Registers	Usage
$v0	Contains address of allocated memory
$a0	The number of bytes to be allocated
$s0	Used to store input along with output
$s1	Used to store input along with output
$s2	Used to store user input along with output
$s3	Used to store input along with output
$s4	Used to store input along with output
$s5	Used to store input along with output
$s6	Used to store input along with output
$t0	Primarily used to store variables, sums, products, differences, and quotients between registers
$t1	Primarily used to store variables, sums, products, differences, and quotients between registers
$t2	Primarily used to store variables, sums, products, differences, and quotients between registers

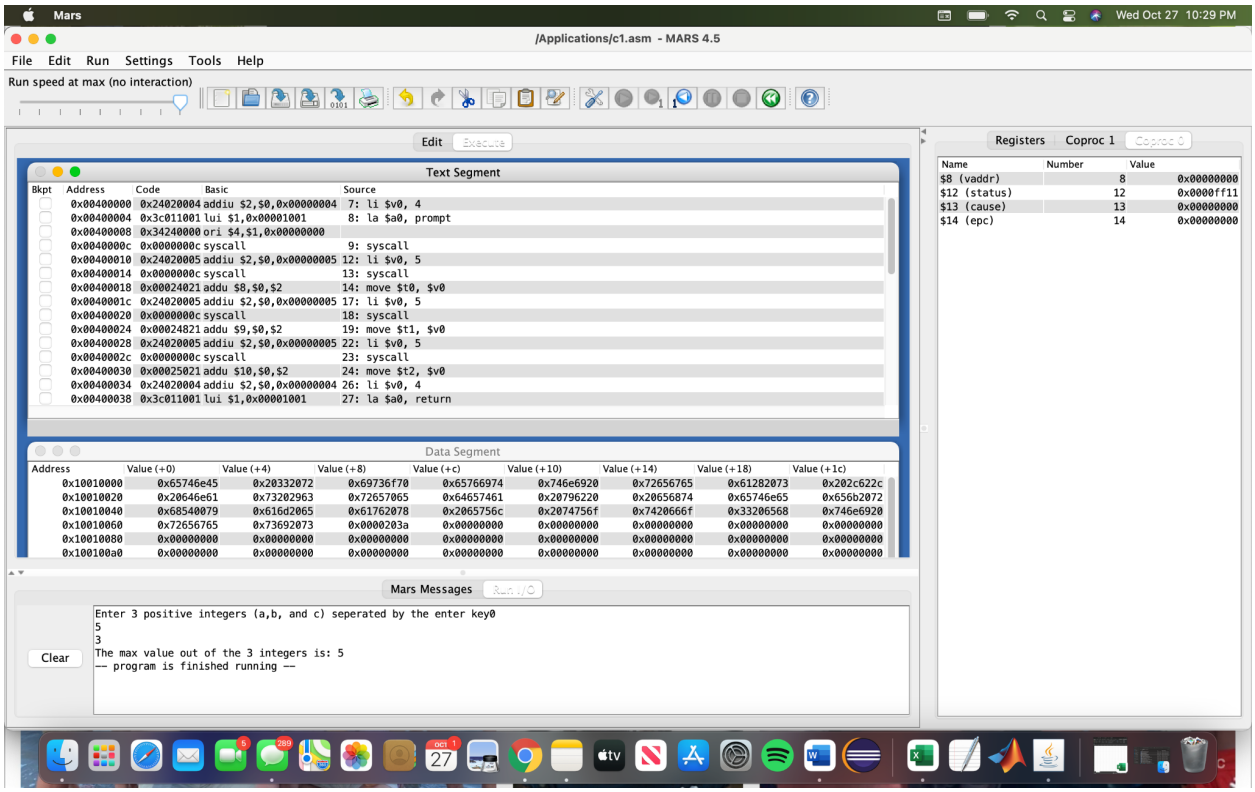
$t3	Primarily used to store variables, sums, products, differences, and quotients between registers
$f1	Single precision floating point register
$f2	Single precision floating point register
$f3	Single precision floating point register
$f4	Single precision floating point register
$f5	Single precision floating point register
$f6	Single precision floating point register

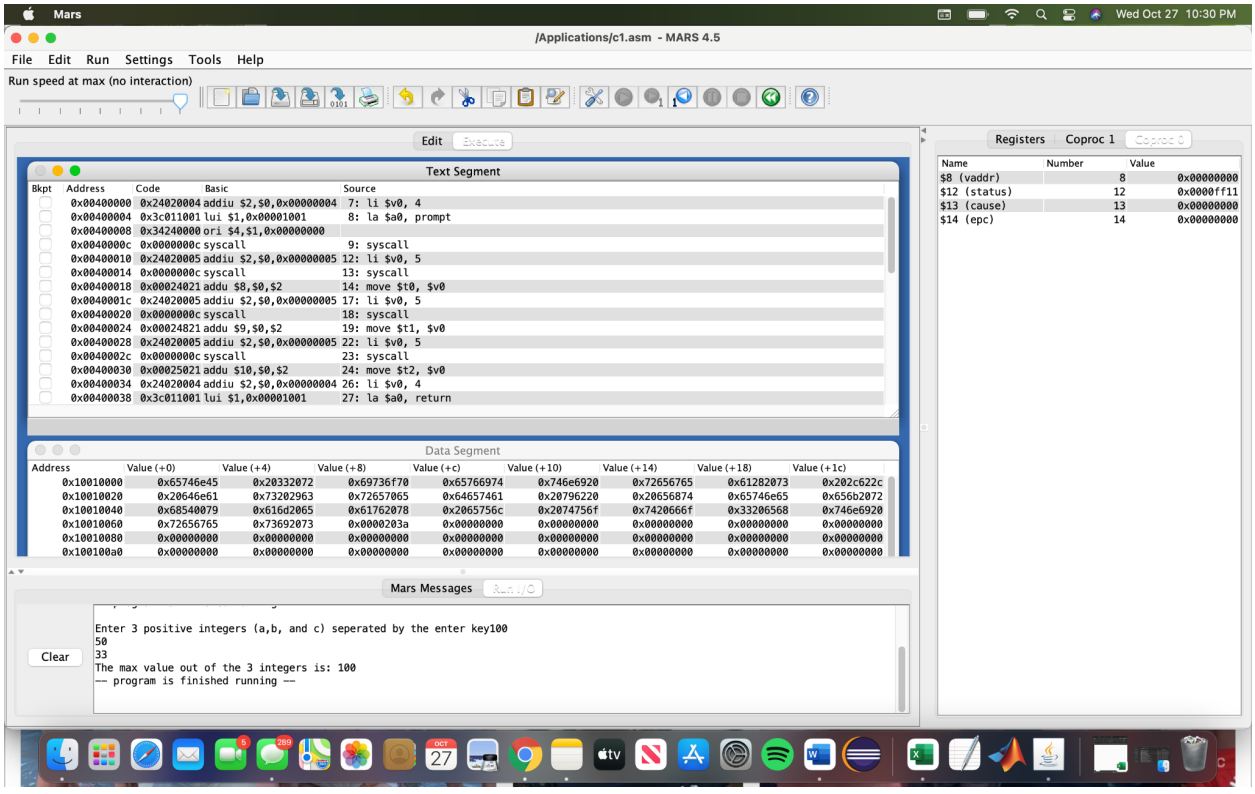
4.0 Learning Coverage

1. Learned how to utilize a stack in order to store data in specific amounts of bits within MIPS
2. How to utilize floating point registers effectively in MIPS in order to conduct math operations
3. The ability to utilize the jump unconditionally function in order travel between functions and registers
4. When to use the jump and link command in order to connect a function to another part of the assembly code
5. How to compare two different values in order to determine which one is greater than or less than and use that to determine where to go next

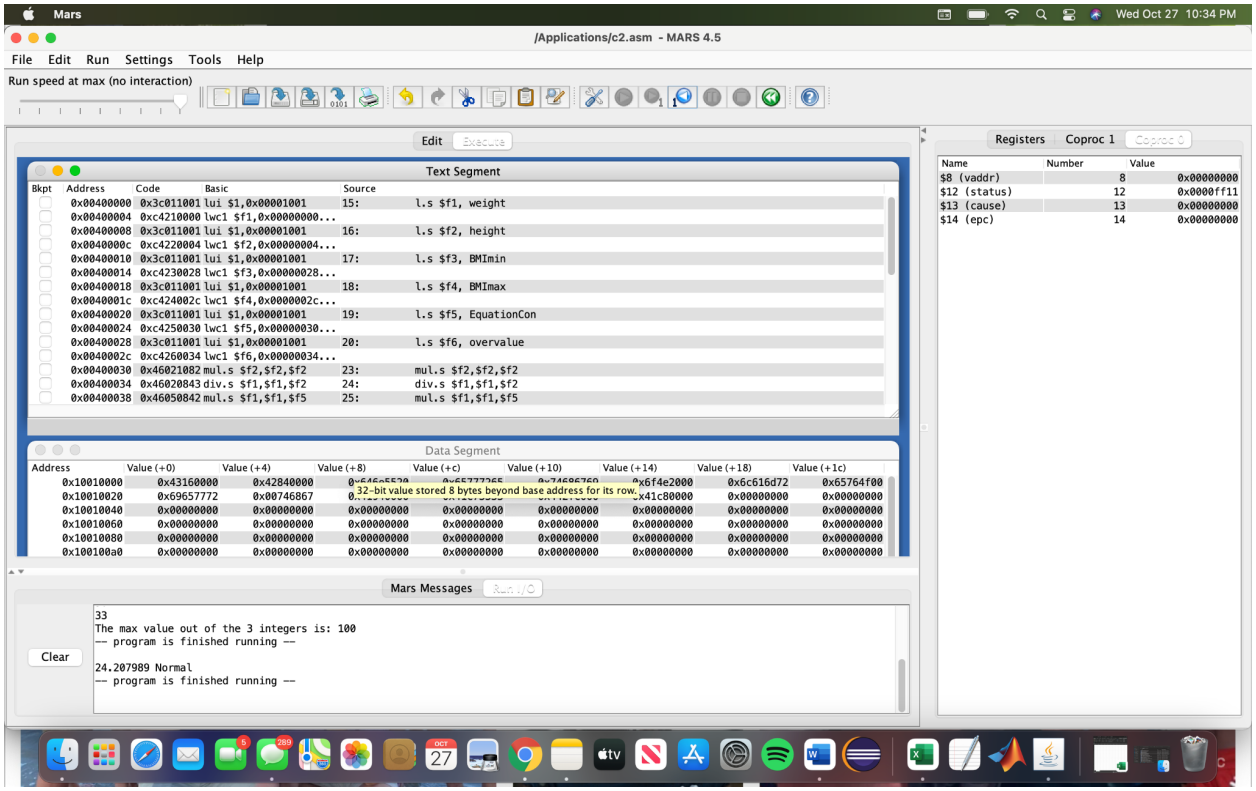
5.0 Test Results

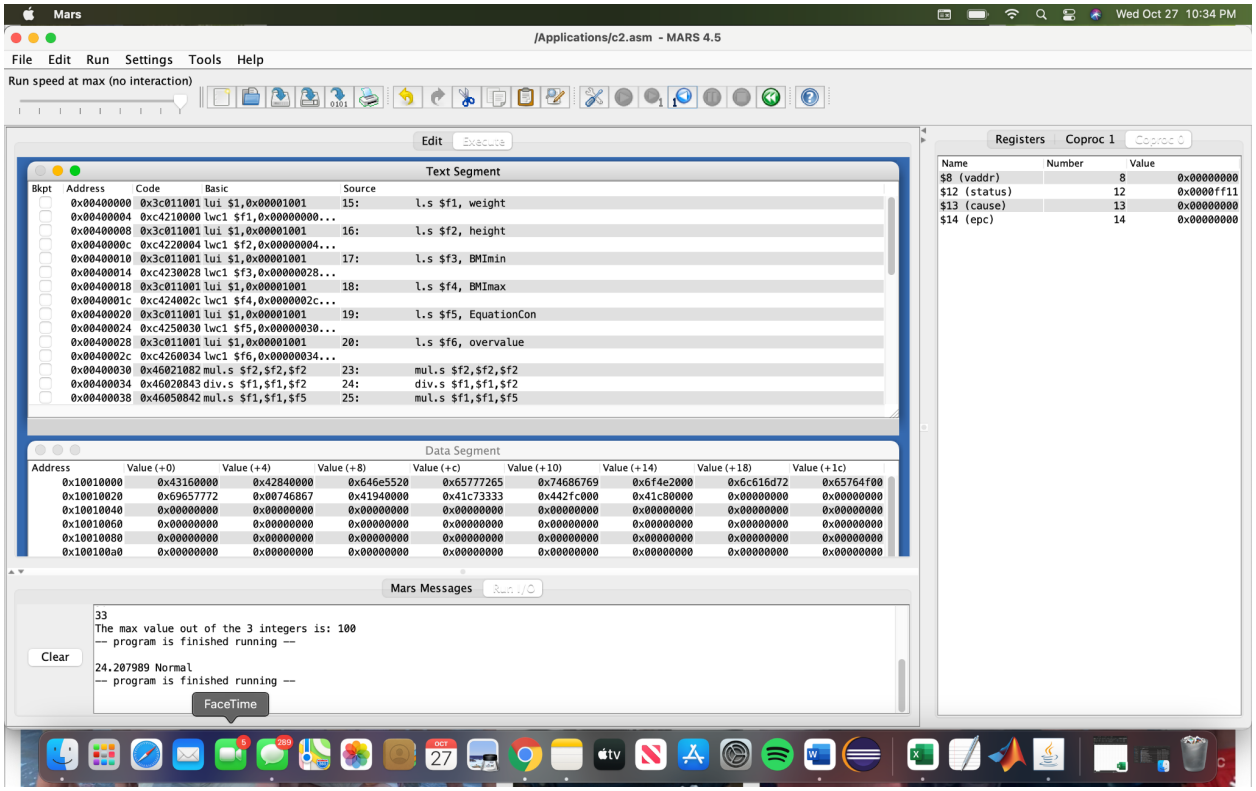
Program 1



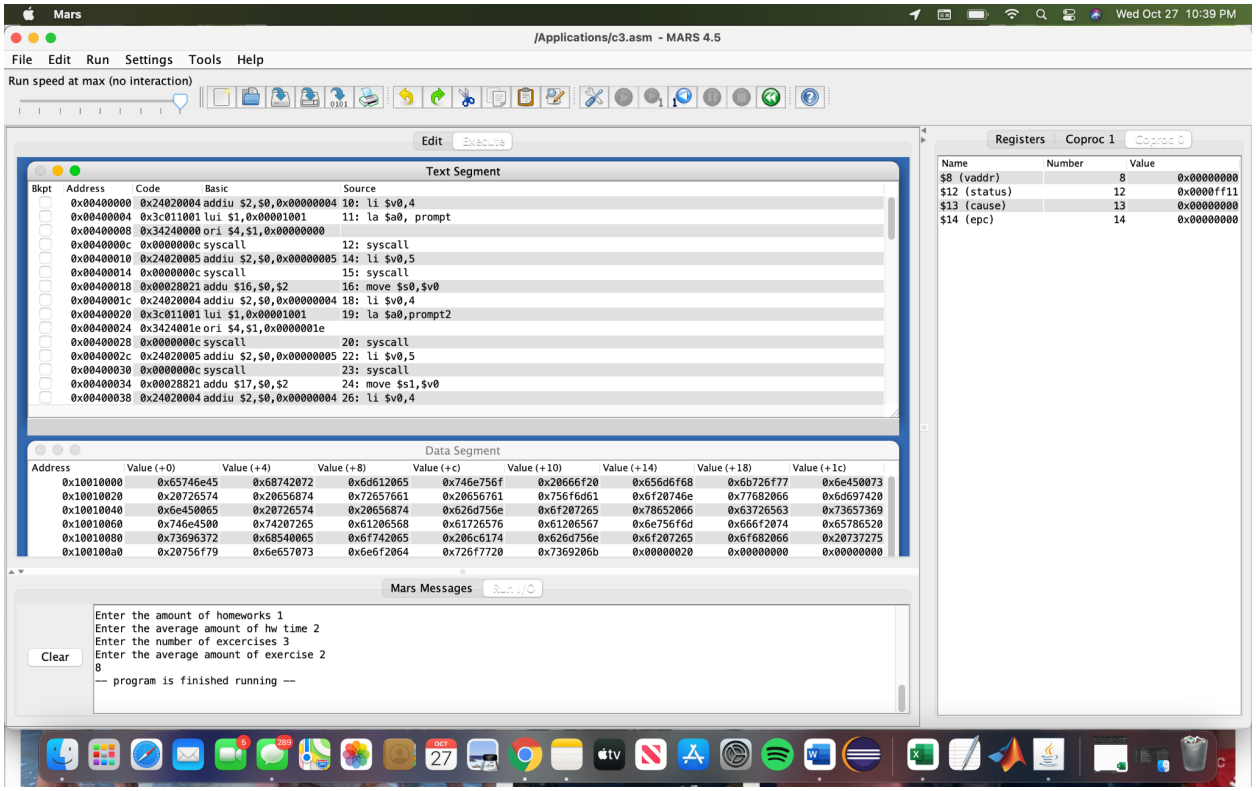


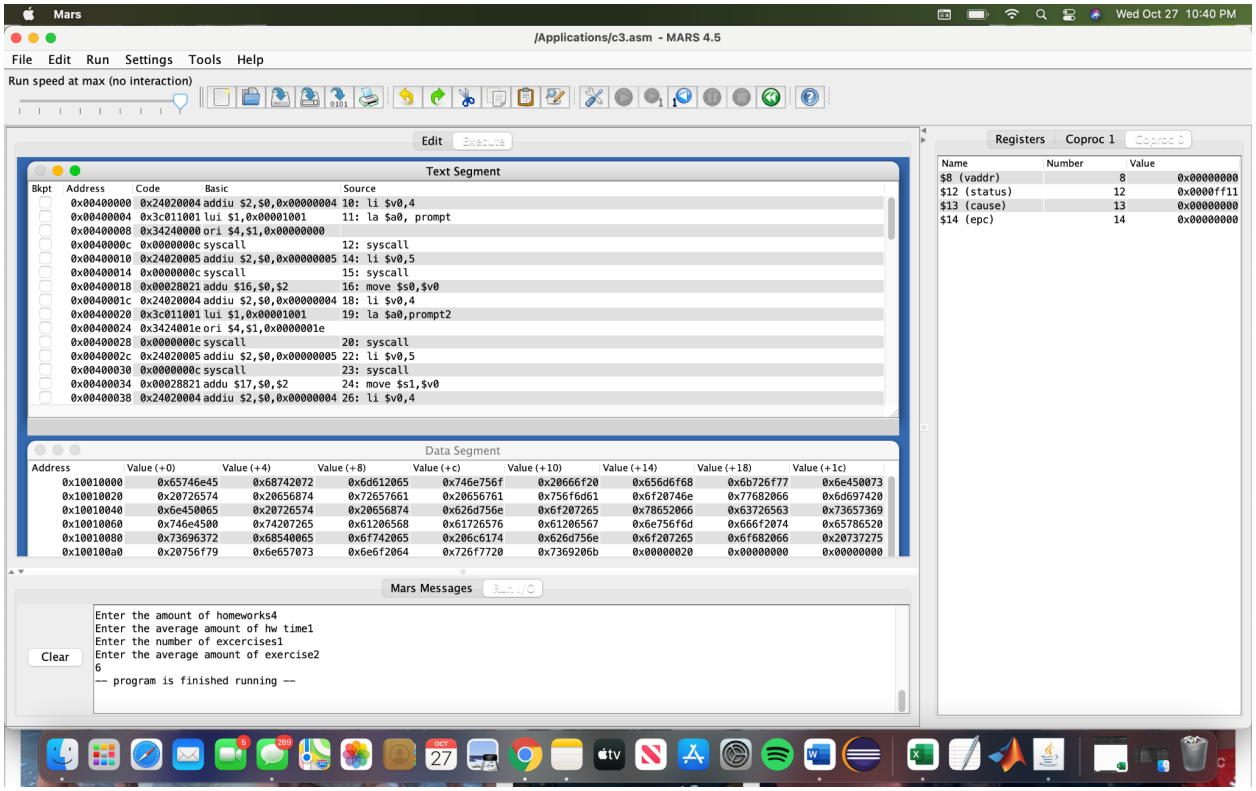
Program 2


Mars
Wed Oct 27 10:34 PM
/Applications/c2.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 0x3c011001 lui $1,0x00001001 15: l.s $f1, weight
0x00400004 0xc4210000 lwc1 $f1,0x00000000...
0x00400008 0x3c011001 lui $1,0x00001001 16: l.s $f2, height
0x0040000c 0xc4220004 lwc1 $f2,0x00000004...
0x00400010 0x3c011001 lui $1,0x00001001 17: l.s $f3, BMImin
0x00400014 0xc4230028 lwc1 $f3,0x00000028...
0x00400018 0x3c011001 lui $1,0x00001001 18: l.s $f4, BMImax
0x0040001c 0xc424002c lwc1 $f4,0x0000002c...
0x00400020 0x3c011001 lui $1,0x00001001 19: l.s $f5, EquationCon
0x00400024 0xc4250030 lwc1 $f5,0x00000030...
0x00400028 0x3c011001 lui $1,0x00001001 20: l.s $f6, overvalue
0x0040002c 0xc4260034 lwc1 $f6,0x00000034...
0x00400030 0x46021082 mul.s $f2,$f2,$f2 23: mul.s $f2,$f2,$f2
0x00400034 0x46020843 div.s $f1,$f1,$f2 24: div.s $f1,$f1,$f2
0x00400038 0x46050842 mul.s $f1,$f1,$f5 25: mul.s $f1,$f1,$f5
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+c) Value (+10) Value (+14) Value (+18) Value (+1c)
0x10010000 0x43160000 0x42840000 0x646e5520 0x65777265 0x74686769 0x6f4e2000 0x6c616d72 0x65764f00
0x10010020 0x69657772 0x00746867 0x41940000 0x41c73333 0x4427c000 0x41c80000 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
32-bit value stored 8 bytes beyond base address for its row.
Registers Coproc 1 Coproc 0
Name Number Value
$8 (vaddr) 8 0x00000000
$12 (status) 12 0x0000ff11
$13 (cause) 13 0x00000000
$14 (epc) 14 0x00000000
Mars Messages Run I/O
33
The max value out of the 3 integers is: 100
-- program is finished running --
24.207989 Normal
-- program is finished running --
Clear


Mars
Wed Oct 27 10:34 PM
/Applications/c2.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 0x3c011001 lui $1,0x00001001 15: l.s $f1, weight
0x00400004 0xc4210000 lwc1 $f1,0x00000000...
0x00400008 0x3c011001 lui $1,0x00001001 16: l.s $f2, height
0x0040000c 0xc4220004 lwc1 $f2,0x00000004...
0x00400010 0x3c011001 lui $1,0x00001001 17: l.s $f3, BMImin
0x00400014 0xc4230028 lwc1 $f3,0x00000028...
0x00400018 0x3c011001 lui $1,0x00001001 18: l.s $f4, BMImax
0x0040001c 0xc424002c lwc1 $f4,0x0000002c...
0x00400020 0x3c011001 lui $1,0x00001001 19: l.s $f5, EquationCon
0x00400024 0xc4250030 lwc1 $f5,0x00000030...
0x00400028 0x3c011001 lui $1,0x00001001 20: l.s $f6, overvalue
0x0040002c 0xc4260034 lwc1 $f6,0x00000034...
0x00400030 0x46021082 mul.s $f2,$f2,$f2 23: mul.s $f2,$f2,$f2
0x00400034 0x46020843 div.s $f1,$f1,$f2 24: div.s $f1,$f1,$f2
0x00400038 0x46050842 mul.s $f1,$f1,$f5 25: mul.s $f1,$f1,$f5
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+c) Value (+10) Value (+14) Value (+18) Value (+1c)
0x10010000 0x43160000 0x42840000 0x646e5520 0x65777265 0x74686769 0x6f4e2000 0x6c616d72 0x65764f00
0x10010020 0x69657772 0x00746867 0x41940000 0x41c73333 0x4427c000 0x41c80000 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
Registers Coproc 1 Coproc 0
Name Number Value
$8 (vaddr) 8 0x00000000
$12 (status) 12 0x0000ff11
$13 (cause) 13 0x00000000
$14 (epc) 14 0x00000000
Mars Messages Run I/O
33
The max value out of the 3 integers is: 100
-- program is finished running --
24.207989 Normal
-- program is finished running --
Clear
FaceTime

Program 3


Mars
Wed Oct 27 10:39 PM
/Applications/c3.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 0x24020004 addiu $2,$0,0x00000004 10: li $v0,4
0x00400004 0x3c011001 lui $1,0x00001001 11: la $a0, prompt
0x00400008 0x34240000 ori $4,$1,0x00000000
0x0040000c 0x0000000c syscall 12: syscall
0x00400010 0x24020005 addiu $2,$0,0x00000005 14: li $v0,5
0x00400014 0x0000000c syscall 15: syscall
0x00400018 0x00028021 addu $16,$0,$2 16: move $s0,$v0
0x0040001c 0x24020004 addiu $2,$0,0x00000004 18: li $v0,4
0x00400020 0x3c011001 lui $1,0x00001001 19: la $a0,prompt2
0x00400024 0x3424001e ori $4,$1,0x0000001e
0x00400028 0x0000000c syscall 20: syscall
0x0040002c 0x24020005 addiu $2,$0,0x00000005 22: li $v0,5
0x00400030 0x0000000c syscall 23: syscall
0x00400034 0x00028821 addu $17,$0,$2 24: move $s1,$v0
0x00400038 0x24020004 addiu $2,$0,0x00000004 26: li $v0,4
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+c) Value (+10) Value (+14) Value (+18) Value (+1c)
0x10010000 0x65746e45 0x68742072 0x6d612065 0x746e756f 0x20666f20 0x656d6f68 0x6b726f77 0x6e450073
0x10010020 0x20726574 0x20656874 0x72657661 0x20656761 0x756f6d61 0x6f20746e 0x77682066 0x6d697420
0x10010040 0x6e450065 0x20726574 0x20656874 0x626d756e 0x6f207265 0x78652066 0x63726563 0x73657369
0x10010060 0x746e4500 0x74207265 0x61206568 0x61726576 0x61206567 0x6e756f6d 0x666f2074 0x65786520
0x10010080 0x73696372 0x68540065 0x6f742065 0x206c6174 0x626d756e 0x6f207265 0x6f682066 0x20737275
0x100100a0 0x20756f79 0x6e657073 0x6e6f2064 0x726f7720 0x7369206b 0x00000020 0x00000000 0x00000000
Registers Coproc 1 Coproc 0
Name Number Value
$8 (vaddr) 8 0x00000000
$12 (status) 12 0x0000ff11
$13 (cause) 13 0x00000000
$14 (epc) 14 0x00000000
Mars Messages Run I/O
Clear
Enter the amount of homeworks 1
Enter the average amount of hw time 2
Enter the number of excercises 3
Enter the average amount of exercise 2
8
-- program is finished running --


Mars
Wed Oct 27 10:40 PM
/Applications/c3.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 0x24020004 addiu $2,$0,0x00000004 10: li $v0,4
0x00400004 0x3c011001 lui $1,0x00001001 11: la $a0, prompt
0x00400008 0x34240000 ori $4,$1,0x00000000
0x0040000c 0x0000000c syscall 12: syscall
0x00400010 0x24020005 addiu $2,$0,0x00000005 14: li $v0,5
0x00400014 0x0000000c syscall 15: syscall
0x00400018 0x00028021 addu $16,$0,$2 16: move $s0,$v0
0x0040001c 0x24020004 addiu $2,$0,0x00000004 18: li $v0,4
0x00400020 0x3c011001 lui $1,0x00001001 19: la $a0,prompt2
0x00400024 0x3424001e ori $4,$1,0x0000001e
0x00400028 0x0000000c syscall 20: syscall
0x0040002c 0x24020005 addiu $2,$0,0x00000005 22: li $v0,5
0x00400030 0x0000000c syscall 23: syscall
0x00400034 0x00028821 addu $17,$0,$2 24: move $s1,$v0
0x00400038 0x24020004 addiu $2,$0,0x00000004 26: li $v0,4
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+c) Value (+10) Value (+14) Value (+18) Value (+1c)
0x10010000 0x65746e45 0x68742072 0x6d612065 0x746e756f 0x20666f20 0x656d6f68 0x6b726f77 0x6e450073
0x10010020 0x20726574 0x20656874 0x72657661 0x20656761 0x756f6d61 0x6f20746e 0x77682066 0x6d697420
0x10010040 0x6e450065 0x20726574 0x20656874 0x626d756e 0x6f207265 0x78652066 0x63726563 0x73657369
0x10010060 0x746e4500 0x74207265 0x61206568 0x61726576 0x61206567 0x6e756f6d 0x666f2074 0x65786520
0x10010080 0x73696372 0x68540065 0x6f742065 0x206c6174 0x626d756e 0x6f207265 0x6f682066 0x20737275
0x100100a0 0x20756f79 0x6e657073 0x6e6f2064 0x726f7720 0x7369206b 0x00000020 0x00000000 0x00000000
Registers Coproc 1 Coproc 0
Name Number Value
$8 (vaddr) 8 0x00000000
$12 (status) 12 0x0000ff11
$13 (cause) 13 0x00000000
$14 (epc) 14 0x00000000
Mars Messages Run I/O
Clear
Enter the amount of homeworks4
Enter the average amount of hw time1
Enter the number of excercises1
Enter the average amount of exercise2
6
-- program is finished running --