Ean Dodge

```
1.  .macro print (%problem)
2.  li $v0 4
3.      la $a0 %problem         #Macro to print
4.      syscall
5.      .end_macro
6.  .data
7.  intro: .asciiz "Give me three integers, i will give back the two biggest ones\n"
8.  endl: .asciiz "\n"
9.  x: .space 10
10. y: .space 10
11. z: .space 10
12. .text
13.
14. print (intro)
15. li $v0 5
16. la $a0 x
17. syscall         #get x, put in s0
18. add $s0, $v0, $zero
19.
20. li $v0 5
21. la $a0 y            #get y, put in s1
22. syscall
23. add $s1, $v0, $zero
24.
25. li $v0 5
26. la $a0 z
27. syscall             #get z, put in s2
28. add $s2, $v0, $zero
29.
30. #x-> $s0
31. #y-> $s1
32. #z-> $s2
33.
34. bge $s0, $s1, xgy  #x>y
35. bge $s1, $s0, ygx #y>x
36.
37. xgy:
38. bge $s1, $s2, pxy #x>y>z
39. j pxz           #x>z>y
40.
41. ygx:
42. bge $s2, $s0, pyz #y>z>x
43. j pxy           #y>x>z
```

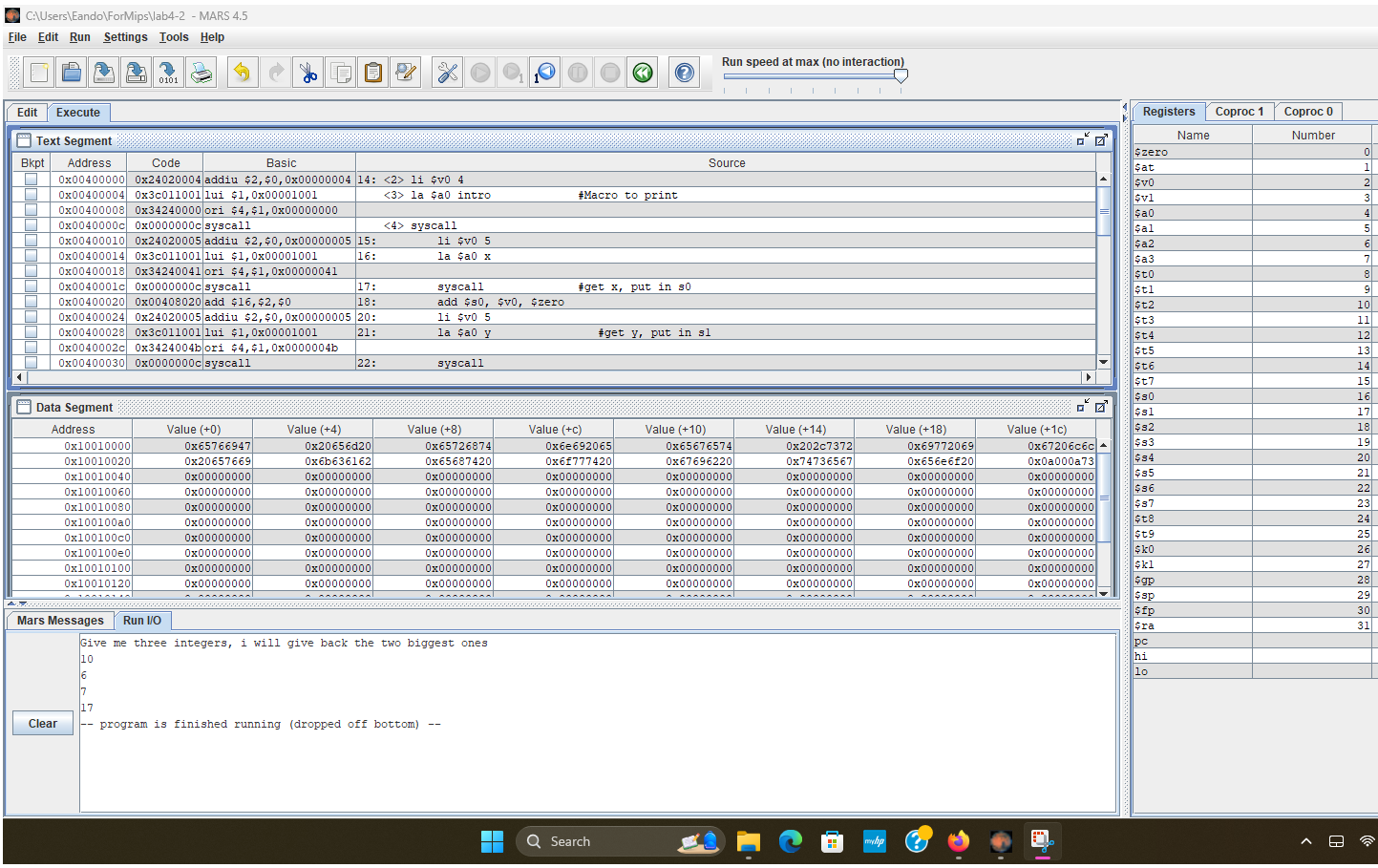
```
44.
45. pxy:
46. add $t0, $s0, $s1
47. add $a0, $t0, $zero
48. li $v0 1           #print out x+y
49. syscall
50. j exit     #go to exit
51.
52. pxz:
53. add $t0, $s0, $s2
54. add $a0, $t0, $zero     #print out x+z
55. li $v0 1
56. syscall
57. j exit     #go to exit
58.
59. pyz:
60. add $t0, $s1, $s2
61. add $a0, $t0, $zero      #print out y+z
62. li $v0 1
63. syscall
64. j exit      #go to exit
65.
66. exit: #do nothing
67.
68.
```

Brief Description:

The first thing I do is I get inputs for x y and z and put them into s0, s1, and s2 respectively. My algorithm for 2 out of three numbers goes as follows. If x is greater than y, jump to xgy(x greater than y), then figure is y > z. If so, then print x plus y, if not, x and z are the biggest, add those and print. Then the opposite, at the beginning, if y is greater than x, then jump to ygx(y greater than x), then figure if x is greater than z, if so, y and x are the biggest so add them and print. If x is not greater than z, then y and z are the biggest, so add them and print. The acronyms I use, like pxy means print x+y.

Conclusion:

I had a lot of fun putting this algorithm together, I learned that I have to jump to an exit. I kept running it and it would run all of pxy, pyz, pxz.


C:\Users\Eando\ForMips\lab4-2 - 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 14: <2> li $v0 4
0x00400004 0x3c011001 lui $1,0x00001001 <3> la $a0 intro #Macro to print
0x00400008 0x34240000 ori $4,$1,0x00000000
0x0040000c 0x0000000c syscall <4> syscall
0x00400010 0x24020005 addiu $2,$0,0x00000005 15: li $v0 5
0x00400014 0x3c011001 lui $1,0x00001001 16: la $a0 x
0x00400018 0x34240041 ori $4,$1,0x00000041
0x0040001c 0x0000000c syscall 17: syscall #get x, put in s0
0x00400020 0x00408020 add $16,$2,$0 18: add $s0, $v0, $zero
0x00400024 0x24020005 addiu $2,$0,0x00000005 20: li $v0 5
0x00400028 0x3c011001 lui $1,0x00001001 21: la $a0 y #get y, put in s1
0x0040002c 0x3424004b ori $4,$1,0x0000004b
0x00400030 0x0000000c syscall 22: syscall
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+c) Value (+10) Value (+14) Value (+18) Value (+1c)
0x10010000 0x65766947 0x20656d20 0x65726874 0x6e692065 0x65676574 0x202c7372 0x69772069 0x67206c6c
0x10010020 0x20657669 0x6b636162 0x65687420 0x6f777420 0x67696220 0x74736567 0x656e6f20 0x0a000a73
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
Registers Coproc 1 Coproc 0
Name Number
$zero 0
$at 1
$v0 2
$v1 3
$a0 4
$a1 5
$a2 6
$a3 7
$t0 8
$t1 9
$t2 10
$t3 11
$t4 12
$t5 13
$t6 14
$t7 15
$s0 16
$s1 17
$s2 18
$s3 19
$s4 20
$s5 21
$s6 22
$s7 23
$t8 24
$t9 25
$k0 26
$k1 27
$gp 28
$sp 29
$fp 30
$ra 31
pc
hi
lo
Mars Messages Run I/O
Clear
Give me three integers, i will give back the two biggest ones
10
6
7
17
-- program is finished running (dropped off bottom) --
Search