Operating systems Assignment 1 Report

Question 1.

The question is asking you to calculate a simple expression derived from the last 7 digits of the UCD student number

My student number is 22203536. Therefore, my expression is 2*0/5/6 = 0.

I loaded the numbers

2 into t0

0 into t1

5 into t2

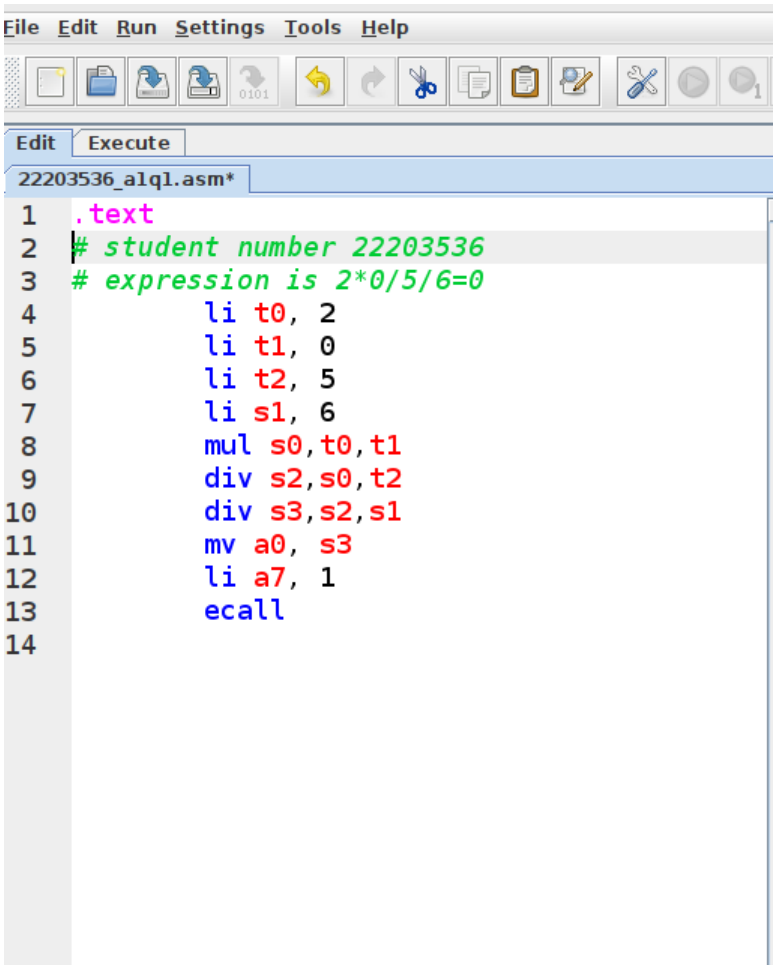
6 into s1

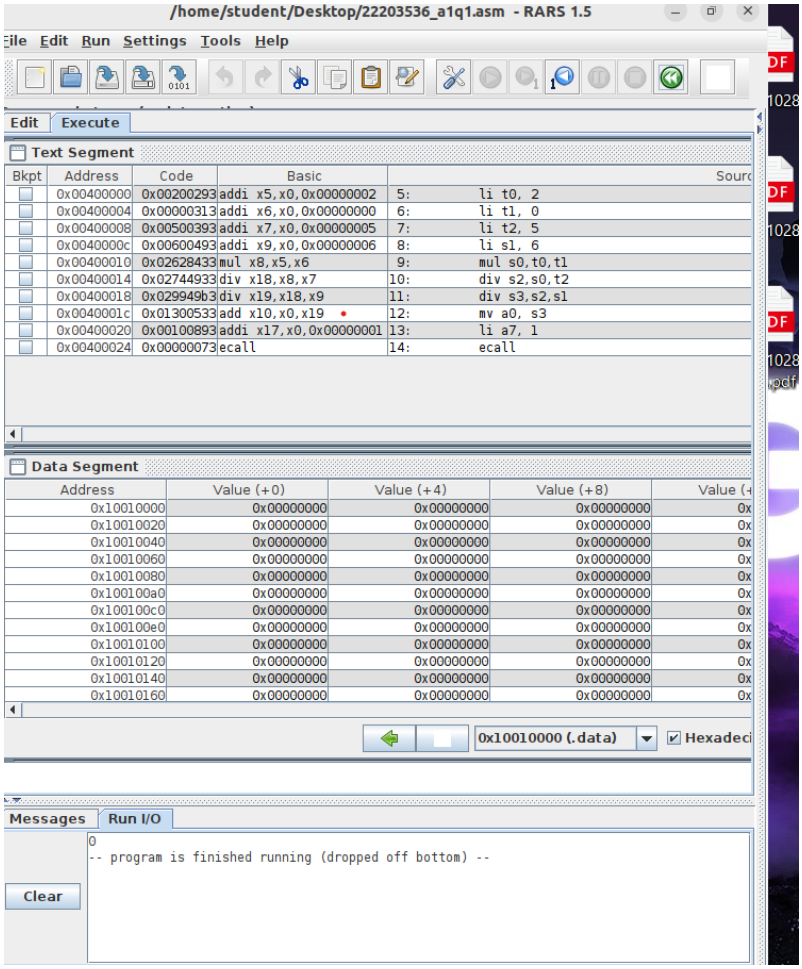
I then multiplied what was in t0 and t1 using mul and stored it in s0. I then divided by the value in t2 and stored in in s2. I then divided by the value in s1 which was 6 and entered the answer into s3.

I moved the value from s3 into a0 and then printed the value by loading 1 into a7 and performing the ecall.

The value that printed was 0 which was correct.

My code may be viewed in the relevant asm file. I enclose a screenshot of my code and output below.


File Edit Run Settings Tools Help
Edit Execute
22203536_a1q1.asm*
1 .text
2 # student number 22203536
3 # expression is 2*0/5/6=0
4 li t0, 2
5 li t1, 0
6 li t2, 5
7 li s1, 6
8 mul s0,t0,t1
9 div s2,s0,t2
10 div s3,s2,s1
11 mv a0, s3
12 li a7, 1
13 ecall
14



Question 2

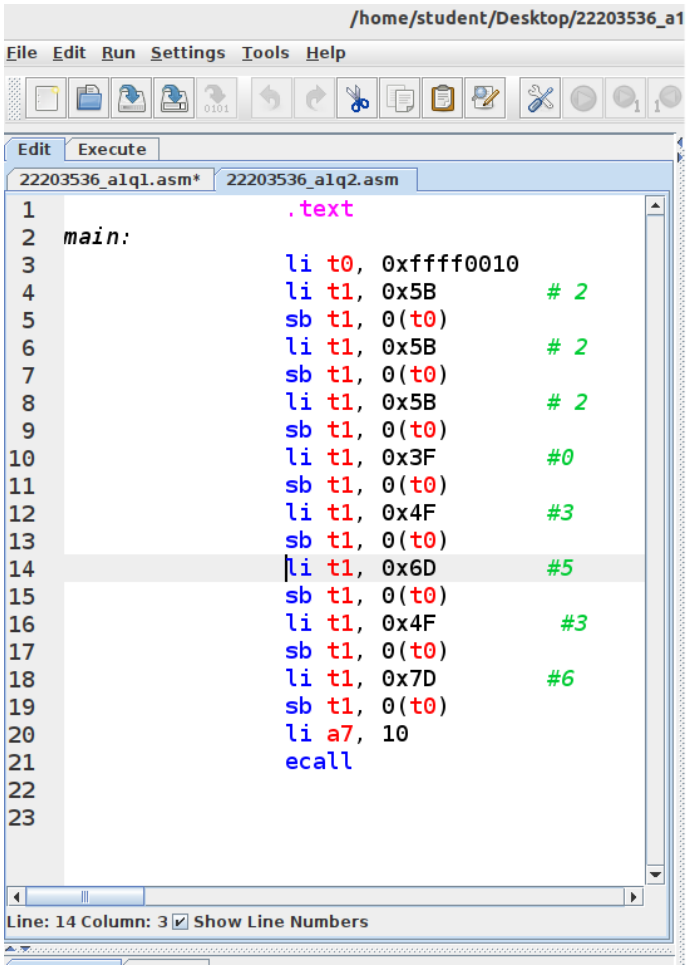
For this question I had to look up the binary number corresponding to what would cause a number to appear in the digital sim lab from the lecture notes. I had to do this for each of the digits in my student number. I then converted those binary numbers into hexadecimal numbers.

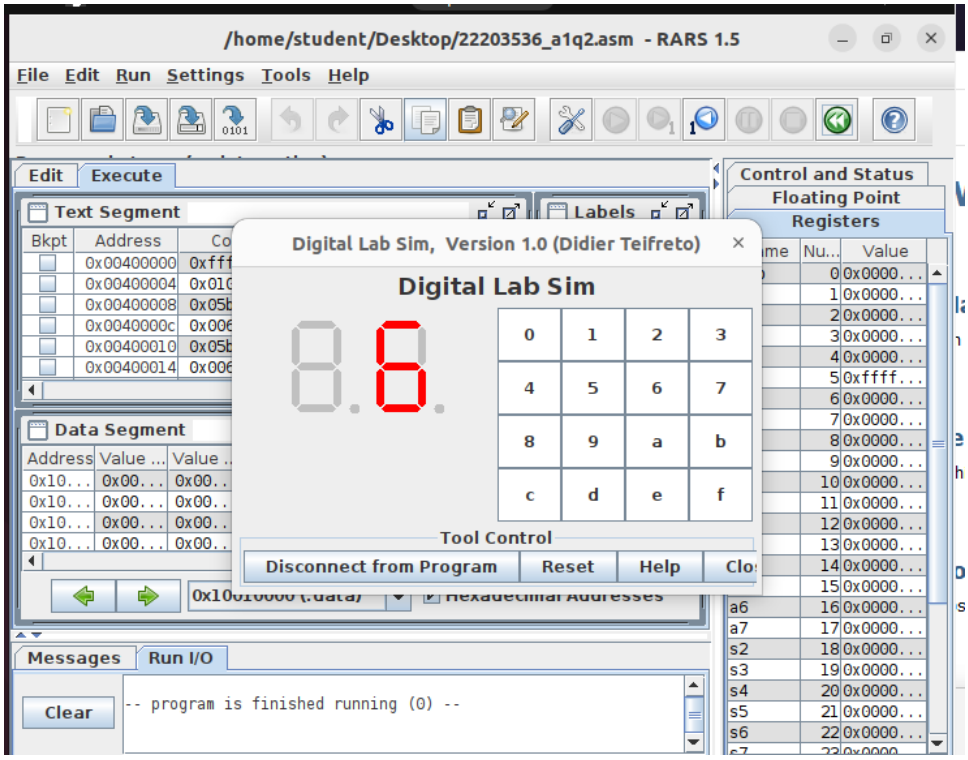
My student number is 22203536. So, I needed the numbers 2 0 3 5 and 6 to appear on the display. When I converted to hexadecimal, I got the following values for 2,0,3,5,6.

5B, 3F, 4F ,6D ,7D.

I then loaded each of these numbers using li and stored using store byte (sb) in t0. I then loaded 10 in a7 and then performed the ecall. I then ran the code and noticed all of the digits from my student

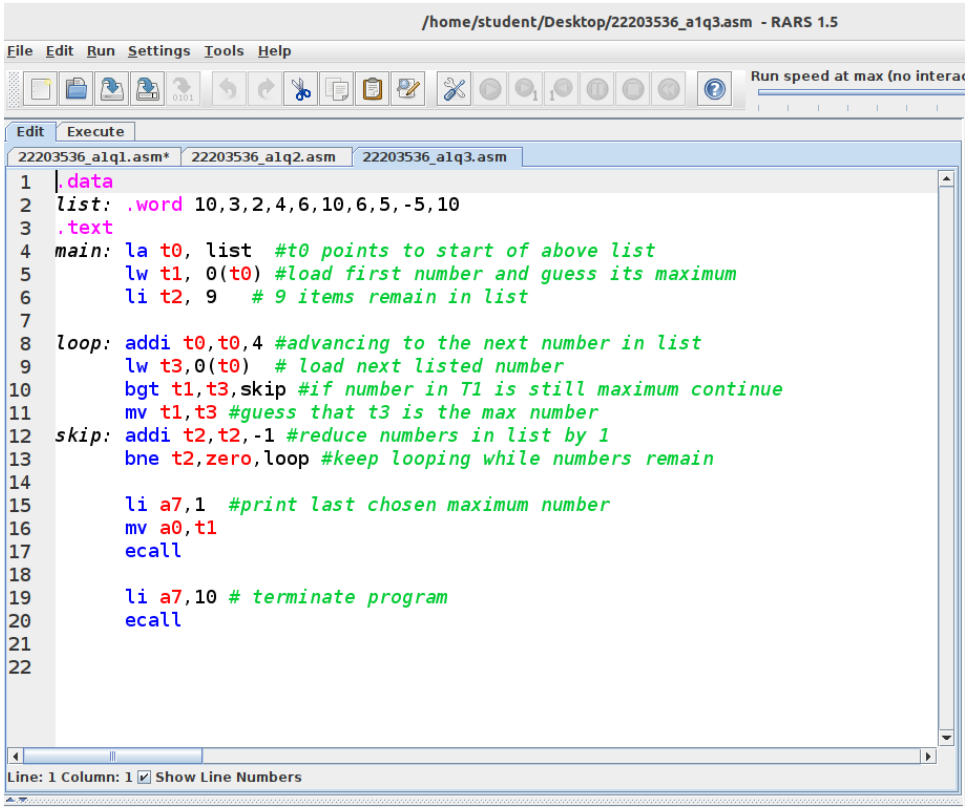
number were working in the digital sim lab. Below I have included my screenshot of the final state of the program in addition to a screenshot of my code. My code may also be viewed in the asm file. It displays the last digit of my student number.

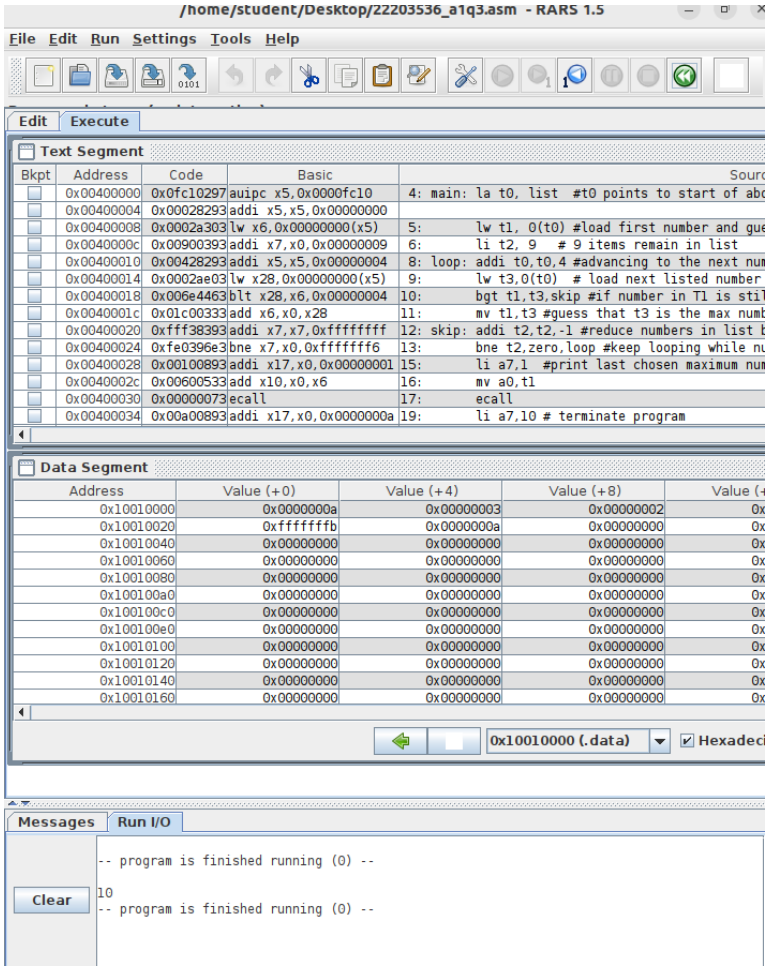




Question 3

This question required you to change branch if less than to branch if greater than. No other changes were necessary in order to get the code to find the maximum number instead of the minimum number in the list. I ran the code and the output was indeed the maximum number in the list. My code may be viewed in the relevant asm file. Below I have included a screenshot of the code and output.


/home/student/Desktop/22203536_a1q3.asm - RARS 1.5
File Edit Run Settings Tools Help
Run speed at max (no interac
Edit Execute
22203536_a1q1.asm*
22203536_a1q2.asm
22203536_a1q3.asm
1 .data
2 list: .word 10,3,2,4,6,10,6,5,-5,10
3 .text
4 main: la t0, list #t0 points to start of above list
5 lw t1, 0(t0) #load first number and guess its maximum
6 li t2, 9 # 9 items remain in list
7
8 loop: addi t0,t0,4 #advancing to the next number in list
9 lw t3,0(t0) # load next listed number
10 bgt t1,t3,skip #if number in T1 is still maximum continue
11 mv t1,t3 #guess that t3 is the max number
12 skip: addi t2,t2,-1 #reduce numbers in list by 1
13 bne t2,zero,loop #keep looping while numbers remain
14
15 li a7,1 #print last chosen maximum number
16 mv a0,t1
17 ecall
18
19 li a7,10 # terminate program
20 ecall
21
22
Line: 1 Column: 1 Show Line Numbers


/home/student/Desktop/22203536_a1q3.asm - RARS 1.5
File Edit Run Settings Tools Help
Edit Execute
Text Segment
Bkpt Address Code Basic Sourc
0x00400000 0x0fc10297 auipc x5,0x0000fc10 4: main: la t0, list #t0 points to start of abo
0x00400004 0x00028293 addi x5,x5,0x00000000
0x00400008 0x0002a303 lw x6,0x00000000(x5) 5: lw t1, 0(t0) #load first number and gue
0x0040000c 0x00900393 addi x7,x0,0x00000009 6: li t2, 9 # 9 items remain in list
0x00400010 0x00428293 addi x5,x5,0x00000004 8: loop: addi t0,t0,4 #advancing to the next num
0x00400014 0x0002ae03 lw x28,0x00000000(x5) 9: lw t3,0(t0) # load next listed number
0x00400018 0x006e4463 blt x28,x6,0x00000004 10: bgt t1,t3,skip #if number in T1 is stil
0x0040001c 0x01c00333 add x6,x0,x28 11: mv t1,t3 #guess that t3 is the max numb
0x00400020 0xfff38393 addi x7,x7,0xffffffff 12: skip: addi t2,t2,-1 #reduce numbers in list l
0x00400024 0xfe0396e3 bne x7,x0,0xfffffff6 13: bne t2,zero,loop #keep looping while nu
0x00400028 0x00100893 addi x17,x0,0x00000001 15: li a7,1 #print last chosen maximum num
0x0040002c 0x00600533 add x10,x0,x6 16: mv a0,t1
0x00400030 0x00000073 ecall 17: ecall
0x00400034 0x00a00893 addi x17,x0,0x0000000a 19: li a7,10 # terminate program
Data Segment
Address Value (+0) Value (+4) Value (+8) Value (+
0x10010000 0x0000000a 0x00000003 0x00000002 0x
0x10010020 0xfffffffb 0x0000000a 0x00000000 0x
0x10010040 0x00000000 0x00000000 0x00000000 0x
0x10010060 0x00000000 0x00000000 0x00000000 0x
0x10010080 0x00000000 0x00000000 0x00000000 0x
0x100100a0 0x00000000 0x00000000 0x00000000 0x
0x100100c0 0x00000000 0x00000000 0x00000000 0x
0x100100e0 0x00000000 0x00000000 0x00000000 0x
0x10010100 0x00000000 0x00000000 0x00000000 0x
0x10010120 0x00000000 0x00000000 0x00000000 0x
0x10010140 0x00000000 0x00000000 0x00000000 0x
0x10010160 0x00000000 0x00000000 0x00000000 0x
0x10010000 (.data) Hexadeci
Messages Run I/O
Clear
-- program is finished running (0) --
10
-- program is finished running (0) --