

# Lab 5

## Problem 1

```
1 addi a0, zero, 79 # give a value to a0
2 addi t1, zero, 9  # condition variable
3 addi t2, zero, 1  # condition variable
4 addi t3, zero, 8  # condition variable
5 loop: ble a0, t3, if # if a0 <= 8:
6 sub a0, a0, t1      # a0 = a0 - 9
7 j loop             # go back to line 5
8 if: bge a0, t2, assign # if a0 >= 1, which means it can not be divisible by 9
9 j end              # if a0 = 9, which means it can be divisible by 9
10 assign: addi a0, zero, 1 # will set it to 1
11 end: add a0, a0, zero   # final output
12
```

## Problem 2

```
1 lui s0, 0xA1A2A # s0 = 0xA1A2A000
2 addi s0, s0, 0x3A4 # s0 = 0xA1A2A3A4
3 sw s0, 0x300(zero) # memory[300] = 0xA1A2A3A4
4
5 lui s0, 0xB1B2B # s0 = 0xB1B2B000
6 addi s0, s0, 0x3B4 # s0 = 0xB1B2B3B4
7 sw s0, 0x304(zero) # memory[304] = 0xB1B2B3B4
8
9 lui s0, 0xC1C2C # s0 = 0xC1C2C000
10 addi s0, s0, 0x3C4 # s0 = 0xC1C2C3C4
11 sw s0, 0x308(zero) # memory[308] = 0xC1C2C3C4
12
13 lui s0, 0xD1D2D # s0 = 0xD1D2D000
14 addi s0, s0, 0x3D4 # s0 = 0xD1D2D3D4
15 sw s0, 0x30c(zero) # memory[30c] = 0xD1D2D3D4
16
17 lui s0, 0xE1E2E # s0 = 0xE1E2E000
18 addi s0, s0, 0x3E4 # s0 = 0xE1E2E3E4
19 sw s0, 0x310(zero) # memory[310] = 0xE1E2E3E4
20
21 lui s0, 0xF1F2F # s0 = 0xF1F2F000
22 addi s0, s0, 0x3F4 # s0 = 0xF1F2F3F4
23 sw s0, 0x314(zero) # memory[314] = 0xF1F2F3F4
24
25 lui s0, 0x11223 # s0 = 0x11223000
26 addi s0, s0, 0x344 # s0 = 0x11223344
27 sw s0, 0x318(zero) # memory[318] = 0x11223344
28
29 lui s0, 0x55667 # s0 = 0x55667000
30 addi s0, s0, 0x788 # s0 = 0x55667788
31 sw s0, 0x31c(zero) # memory[31c] = 0x55667788
```

```

33 lw t0, 0x300(zero) # t0 = 0xA1A2A3A4
34 srli t1, t0, 24 # t1 = 0x000000a1
35
36 slli t2, t0, 8 # t2 = 0xa2a3a400
37 srli t2, t2, 24 # t2 = 0x000000a2
38 slli t2, t2, 8 # t2 = 0x0000a200
39
40 srli t3, t0, 8 # t3 = 0x00a1a2a3
41 slli t3, t3, 24 # t3 = 0xa3000000
42 srli t3, t3, 8 # t3 = 0x00a30000
43
44 slli t4, t0, 24 # t4 = 0xa4000000
45
46 add t5, t1, t2 # t5 = 0x0000a2a1
47 add t5, t5, t3 # t5 = 0x00a3a2a1
48 add t5, t5, t4 # t5 = 0xa4a3a2a1
49 sw t5, 0x300(zero) # memory[300] = t5 = 0xa4a3a2a1
50
51 # The code below will be pretty much like the 300 address, so no comment
52
53 lw t0, 0x304(zero)
54 srli t1, t0, 24
55 slli t2, t0, 8
56 srli t2, t2, 24
57 slli t2, t2, 8
58 srli t3, t0, 8
59 slli t3, t3, 24
60 srli t3, t3, 8
61 slli t4, t0, 24
62 add t5, t1, t2
63 add t5, t5, t3
64 add t5, t5, t4
65 sw t5, 0x304(zero)
66
67 lw t0, 0x308(zero)
68 srli t1, t0, 24
69 slli t2, t0, 8
70 srli t2, t2, 24
71 slli t2, t2, 8
72 srli t3, t0, 8
73 slli t3, t3, 24
74 srli t3, t3, 8
75 slli t4, t0, 24
76 add t5, t1, t2

```

```
77 add t5, t5, t3
78 add t5, t5, t4
79 sw t5, 0x308(zero)
80
81 lw t0, 0x30c(zero)
82 srli t1, t0, 24
83 slli t2, t0, 8
84 srli t2, t2, 24
85 slli t2, t2, 8
86 srli t3, t0, 8
87 slli t3, t3, 24
88 srli t3, t3, 8
89 slli t4, t0, 24
90 add t5, t1, t2
91 add t5, t5, t3
92 add t5, t5, t4
93 sw t5, 0x30c(zero)
94
95 lw t0, 0x310(zero)
96 srli t1, t0, 24
97 slli t2, t0, 8
98 srli t2, t2, 24
99 slli t2, t2, 8
100 srli t3, t0, 8
101 slli t3, t3, 24
102 srli t3, t3, 8
103 slli t4, t0, 24
104 add t5, t1, t2
105 add t5, t5, t3
106 add t5, t5, t4
107 sw t5, 0x310(zero)
108
109 lw t0, 0x314(zero)
110 srli t1, t0, 24
111 slli t2, t0, 8
112 srli t2, t2, 24
113 slli t2, t2, 8
114 srli t3, t0, 8
115 slli t3, t3, 24
116 srli t3, t3, 8
117 slli t4, t0, 24
118 add t5, t1, t2
119 add t5, t5, t3
120 add t5, t5, t4
121 sw t5, 0x314(zero)
```

```
123 lw t0, 0x318(zero)
124 srli t1, t0, 24
125 slli t2, t0, 8
126 srli t2, t2, 24
127 slli t2, t2, 8
128 srli t3, t0, 8
129 slli t3, t3, 24
130 srli t3, t3, 8
131 slli t4, t0, 24
132 add t5, t1, t2
133 add t5, t5, t3
134 add t5, t5, t4
135 sw t5, 0x318(zero)
136
137 lw t0, 0x31c(zero)
138 srli t1, t0, 24
139 slli t2, t0, 8
140 srli t2, t2, 24
141 slli t2, t2, 8
142 srli t3, t0, 8
143 slli t3, t3, 24
144 srli t3, t3, 8
145 slli t4, t0, 24
146 add t5, t1, t2
147 add t5, t5, t3
148 add t5, t5, t4
149 sw t5, 0x31c(zero)
150
```

## Problem 3

```

1 addi s0, zero, 89
2 sw s0, 0x400(zero) # memory[400] = 89
3 addi s0, zero, 63
4 sw s0, 0x404(zero) # memory[404] = 63
5 addi s0, zero, -55
6 sw s0, 0x408(zero) # memory[408] = -55
7 addi s0, zero, -107
8 sw s0, 0x40c(zero) # memory[40c] = -107
9 addi s0, zero, 42
10 sw s0, 0x410(zero) # memory[410] = 42
11 addi s0, zero, 98
12 sw s0, 0x414(zero) # memory[414] = 98
13 addi s0, zero, -425
14 sw s0, 0x418(zero) # memory[418] = -425
15 addi s0, zero, 203
16 sw s0, 0x41c(zero) # memory[41c] = 203
17 addi s0, zero, 0
18 sw s0, 0x420(zero) # memory[420] = 0
19 addi s0, zero, 303
20 sw s0, 0x424(zero) # memory[424] = 303
21
22 addi s0, zero, 1
23 addi t0, zero, 1 # change = 1
24
25 loop: beq t0, zero, end
26 addi t0, zero, 0 # change = 0
27 addi t2, zero, 0 # x = 0
28 addi t3, zero, 4 # y = 1
29 addi t1, zero, 9 # num = 9
30
31 for: beq t1, zero, switch
32 lw s1, 0x400(t2) # s1 = array[x]
33 lw s2, 0x400(t3) # s2 = array[y]
34 blt s1, s2, con # if s1 < s2, then do nothing
35 addi t5, s1, 0 # otherwise: t5 = s1
36 addi s1, s2, 0 # s1 = s2
37 addi s2, t5, 0 # s2 = t5
38 addi t0, t0, 1 # change += 1
39 con: sw s1, 0x400(t2) # array[x] = s1
40 sw s2, 0x400(t3) # array[y] = s2
41 addi t2, t2, 4 # x += 1
42 addi t3, t3, 4 # y += 1
43 addi t1, t1, -1 # t1 -= 1
44 j for
45 switch:
46 j loop
47 end:

```