

This program gets data from memory and doubles the value, then stores it back into memory. This program ends once \$3 becomes greater than \$1.

---

#### Our ISA Simulator

-----

Group 2's Simulator

\*\*\*\*\* Simulation Start \*\*\*\*\*

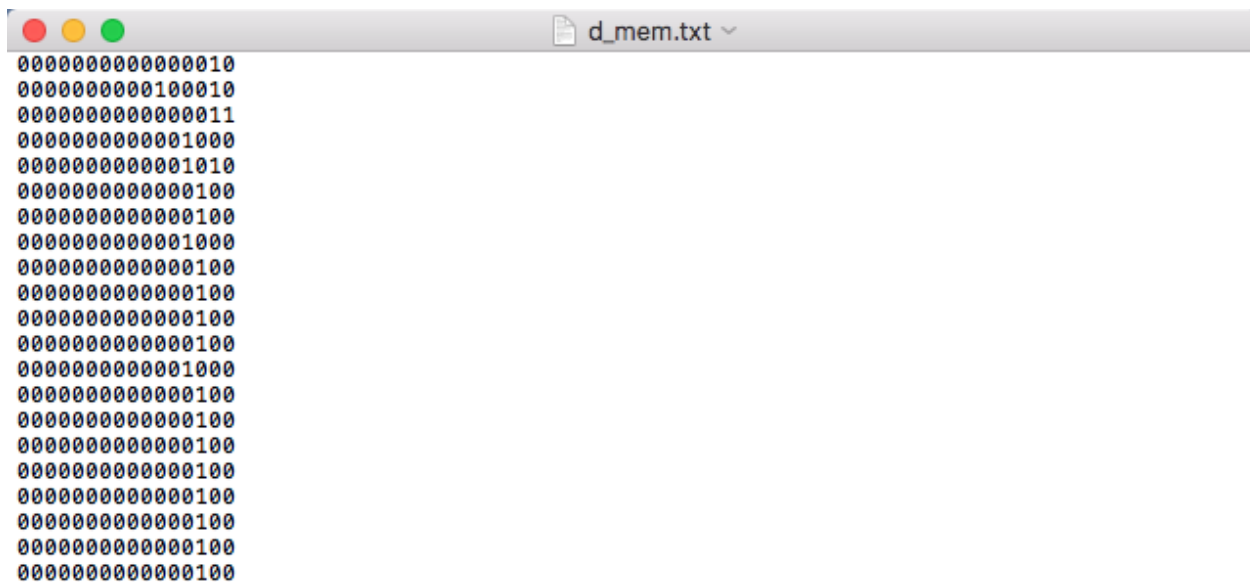
```
li $0, 3
add $0, $0
add $0, $0
lw $3, ($0)
li $1, 1
lw $2, ($1)
add $2, $2
sw $2, ($1)
li $0, 3
add $1, $0
sltR0 $1, $3
beqz $0, $1
li $0, -3
add $0, $0
add $0, $0
beqz $0, $1
beqz $0, $1
jump $0
lw $2, ($1)
add $2, $2
sw $2, ($1)
li $0, 3
add $1, $0
sltR0 $1, $3
beqz $0, $1
li $0, -3
add $0, $0
add $0, $0
beqz $0, $1
beqz $0, $1
jump $0
lw $2, ($1)
add $2, $2
sw $2, ($1)
li $0, 3
add $1, $0

add $1, $0
sltR0 $1, $3
beqz $0, $1
beqz $0, $0
***** Simulation finished *****
Dynamic Instr Count: 39
Registers R0-R3: [0, 10, 8, 8]
>>>
```

---

Below are two screen shots of d\_mem before and after running the program. As you can see, Mem[1], Mem[4], and Mem[7] are twice as big as they were before.

[illegible]



li R0,3	110 0 011
add R0, R0	010 00 00
add R0, R0	010 00 00
lw R3, (R0)	000 11 00
li R1, 1	110 1 001
lw R2, (R1)	000 10 01
add R2, R2	010 10 10
sw R2, (R1)	001 10 01
li R0, 3	110 0 011
add R1, R0	010 01 00
sltR0 R1, R3	0110 1 11
beqz R0, R1	0111 0 01
li R0, -3	110 0 101
add R0, R0	010 00 00
add R0, R0	010 00 00
beqz R0, R1	0111 0 01
beqz R0, R1	0111 0 01
jump R0	11111 00
Halt	0111000
Halt	0111000
Halt	0111000
Halt	0111000