Part I

X	Y	Z	rwe	imm en	imm va	an en	- a/s	lu en	lf	su en	st	st en	ld en	- r/w	msel	Description
2	-	3	1	1	-1	0	-	0	-	1	01	0	0	-	0	R3 = R2 << 1
3	2	3	1	0	-	1	0	0	-	0	--	0	0	-	0	R3 = R2 + R3
-	-	1	1	1	100	0	-	0	-	0	--	0	1	1	1	R1 = M[100]
1	-	1	1	1	-2	0	-	0	-	1	01	0	0	-	0	R1 = R1 << 2
1	3	1	1	0	-	1	0	0	-	0	--	0	0	-	0	R1 = R3 + R1

Part II

```
.data
# This is the start of the original array.
Original:  .word  200,  270,  250,  100
           .word  205,  230,  105,  235
           .word  190,   95,   90,  205
           .word   80,  205,  110,  215
# The next statement allocates room for the other array.
# The array takes up 4*16=64 bytes.

Second:    .space 64
.align 2
.globl main
.text

# hard coding at its best
# take two diagonally symmetrical numbers, swap
main:  addi  $t0,  $0,  4    # t0 = 4
       addi  $t1,  $0,  16   # t1 = 16
       lw    $t2,  Original($t0)    # t2 = Original(4)
       lw    $t3,  Original($t1)    # t3 = Original(16)
```

```
sw    $t3,  Second($t0)    # Original(4) = t3
sw    $t2,  Second($t1)    # Original(16) = t2

addi  $t0,  $0,  8    # t0 = 4
addi  $t1,  $0,  32   # t1 = 16
lw    $t2,  Original($t0)    # t2 = Original(4)
lw    $t3,  Original($t1)    # t3 = Original(16)
sw    $t3,  Second($t0)    # Original(4) = t3
sw    $t2,  Second($t1)    # Original(16) = t2

addi  $t0,  $0,  12    # t0 = 4
addi  $t1,  $0,  48   # t1 = 16
lw    $t2,  Original($t0)    # t2 = Original(4)
lw    $t3,  Original($t1)    # t3 = Original(16)
sw    $t3,  Second($t0)    # Original(4) = t3
sw    $t2,  Second($t1)    # Original(16) = t2

addi  $t0,  $0,  24    # t0 = 4
addi  $t1,  $0,  36   # t1 = 16
lw    $t2,  Original($t0)    # t2 = Original(4)
lw    $t3,  Original($t1)    # t3 = Original(16)
sw    $t3,  Second($t0)    # Original(4) = t3
sw    $t2,  Second($t1)    # Original(16) = t2

addi  $t0,  $0,  28    # t0 = 4
addi  $t1,  $0,  52   # t1 = 16
lw    $t2,  Original($t0)    # t2 = Original(4)
lw    $t3,  Original($t1)    # t3 = Original(16)
sw    $t3,  Second($t0)    # Original(4) = t3
sw    $t2,  Second($t1)    # Original(16) = t2
```

```
    addi  $t0,  $0,  44    # t0 = 4
    addi  $t1,  $0,  56   # t1 = 16
    lw    $t2,  Original($t0)    # t2 = Original(4)
    lw    $t3,  Original($t1)    # t3 = Original(16)
    sw    $t3,  Second($t0)    # Original(4) = t3
    sw    $t2,  Second($t1)    # Original(16) = t2

    addi  $t0,  $0,  0    # t0 = 4
    addi  $t1,  $0,  20   # t1 = 16
    lw    $t2,  Original($t0)    # t2 = Original(4)
    lw    $t3,  Original($t1)    # t3 = Original(16)
    sw    $t3,  Second($t1)    # Original(4) = t3
    sw    $t2,  Second($t0)    # Original(16) = t2

    addi  $t0,  $0,  40    # t0 = 4
    addi  $t1,  $0,  60   # t1 = 16
    lw    $t2,  Original($t0)    # t2 = Original(4)
    lw    $t3,  Original($t1)    # t3 = Original(16)
    sw    $t3,  Second($t1)    # Original(4) = t3
    sw    $t2,  Second($t0)    # Original(16) = t2
Exit:  li   $v0,  10
    syscall
```