RISHABH C. PATEL

ECE 3056 LAB0

DUE: 24 August 2018

Part I:

Microcode for the expression: R1 = 3R2 + 4M[100]

X	*Y*	*Z*	*rwe*	*imen*	*imva*	*auen*	*~a/s*	*luen*	*lf*	*suen*	*st*	*lden*	*sten*	*r/~w*	*msel*	*description*
X	X	3	1	1	100	0	X	1	1100	0	XX	0	0	X	0	R3 = 100
3	X	3	1	0	X	0	X	0	X	0	XX	1	0	1	1	R3 = M[R3]
3	X	3	1	1	-2	0	X	0	X	1	1	0	0	X	0	R3 = 4R3
2	X	5	1	1	-1	0	X	0	X	1	1	0	0	X	0	R5 = 2R2
5	2	2	1	0	X	1	0	0	X	0	XX	0	0	X	0	R2 = R5 + R2
2	3	1	1	0	X	1	0	0	X	0	XX	0	0	X	0	R1 = R2 + R3

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
main: # Your fully commented program starts here.
      addi $t0, $0, 0        # i = 0
      addi $t2, $0, 4        # counter = 4
      la   $t3, Original
```

```
        la   $t4, Second
Loop1:  beq  $t0, $t2, Exit     # if i = counter, Exit
        addi $t1, $0, 0         # j = 0
Loop2:  beq  $t1, $t2, Inc      # if j = counter, Loop1
        #Operation -> Second[i][j] = Original[j][i]
        mul  $t5, $t0, 4        # 4 * i
        mul  $t6, $t1, 4        # 4 * j
        add  $t5, $t5, $t1      # 4 * i + j
        add  $t6, $t6, $t0      # 4 * j + i
        mul  $t5, $t5, 4        # 4 * (4 * i + j)
        mul  $t6, $t6, 4        # 4 * (4 * j + i)
        lw   $t7, Original($t5) # load Original[4 * (4 * i + j)] in $t7
        sw   $t7, Second($t6)   # Store in $7
        addi $t1, $t1, 1        # increment j
        j Loop2
Inc:    addi $t0, $t0, 1        # increment i
        j Loop1
Exit: li $v0, 10                #terminate program
      syscall
```