```
#Robert Maldonado, CS 2318-003, Assignment 2 Part 1 Program C
#This progam displays the given array of 1, 3, 2, 5, 4 from 1st to 5th
element.
#It then swaps the 4th and 5th, and 2nd and 3rd. Elements in the array.
#It then displays the array backwards from 5th to 1st, with elements
swapped.

                        .data
arr:                    .word 1, 3, 2, 5, 4
firstPrmpt:             .asciiz "\nThe initial array from 1st to 5th are\n"
secondPrmpt:            .asciiz "\nThe final array from 5th to 1st are\n"
arrcomma:               .asciiz ", "

                        .text
                        .globl main
main:
                        #displays the prompt for 1st to 5th
                        li $v0, 4
                        la $a0, firstPrmpt
                        syscall
                        addi $t7, $0, 0
                        addi $t6, $0, 3
                        la $t0, arr

loop1:                  #displays the arr from 1th to 5th
                        li $v0, 1
                        lw $a0, 0($t0)
                        syscall
                        bgt $t7, $t6, exit
                        li $v0, 4
                        la $a0, arrcomma
                        syscall
                        addi $t7, $t7, 1
                        addi $t0, $t0, 4
                        j loop1
exit:
                        #swapping 4th and 5th
                        la $t0, arr
                        lw $t1, 12($t0)
                        lw $t2, 16($t0)
                        sw $t1, 16($t0)
                        sw $t2, 12($t0)

                        #swapping 2nd and 3rd
                        lw $t1, 4($t0)
                        lw $t2, 8($t0)
                        sw $t1, 8($t0)
                        sw $t2, 4($t0)

                        #displaying prompt for 5th to 1st
                        li $v0, 4
                        la $a0, secondPrmpt
                        syscall
                        addi $t7, $0, 0
                        addi $t6, $0, 3
```

```
                        la $t0, arr

loop2:                  #displays the arr from 5th to 1st
                        li $v0, 1
                        lw $a0, 16($t0)
                        syscall
                        bgt $t7, $t6, exit2
                        li $v0, 4
                        la $a0, arrcomma
                        syscall
                        addi $t7, $t7, 1
                        addi $t0, $t0, -4
                        j loop2
exit2:
                        #exits the program
                        li $v0, 10
                        syscall
```