# A program that inputs a 4x4 matrix of single-digit integers one row at a time

# (one row per input line – not one number per line!) and stores it into a

# linear 32-bit Integer Array M

.data

matrixArray: .space 64

inputRow: .asciiz "Input Row "

outputRow: .asciiz "Output Row "

space: .asciiz " "

colon: .asciiz ":"

newLine: .asciiz "\n"

inputIntString: .space 64

inputIntLength: .word 64

.text

.globl main

#Main

main:

jal initializeVariables #Initialize Variables

la $s1, matrixArray

add $t4, $t4, $s1

#Call subroutine to accept userInput and call subroutine to load into array

loop:

beq $t6, $t7, printOutput

jal userInput

jal loadArray

addi $t6, $t6, 1

j loop

#Initialize Variables and call subroutine to display the array

printOutput:

jal initializeVariables

loop2:

beq $t6, $t7, exitProgram

jal outputString

jal printRow

addi $t6, $t6, 1

j loop2

exitProgram:

li $v0,10

syscall

#End Main

#subroutine to print row

printRow:

beq $t0, $t5, endRow

lw $a0, ($s1)

li $v0, 1

syscall

li $v0,4

la $a0,space

syscall

addi $s1, $s1, 4

addi $t0, $t0, 1

j printRow

endRow:

addi $t5, $t5, 4

li $v0,4

la $a0,newLine

syscall

jr $ra

#End subroutine printRow

#subroutine to take userInput

userInput:

li $v0,4

la $a0,inputRow

syscall

li $v0,1

move $a0,$t8

syscall

li $v0,4

la $a0,colon

syscall

li $v0,4

la $a0,space

syscall

addi $t8, $t8, 1

jr $ra

#End subroutine userInput

#subroutine to load userInput into an array

loadArray:

li $v0, 8

la $a0, inputIntString

la $a1, inputIntLength

syscall

li $t2, 0

la $s0, inputIntString

add $t2, $t2, $s0

# Reads one byte at a time, skips when it finds a space char (32)

# Loop exits when found a new line char (10)

while:

lbu $t3, ($t2)

beq $t3, 10, endwhile

beq $t3, 32, foundSpace

addi $t3, $t3, -48

move $t0, $t3

addi $t2, $t2, 1

j while

foundSpace:

sw $t0, ($t4)

li $t0, 0

addi $t4, $t4, 4

addi $t2, $t2, 1

j while

endwhile:

sw $t0, ($t4)

li $t0, 0

addi $t4, $t4, 4

addi $t2, $t2, 1

jr $ra

outputString:

li $v0,4

la $a0,outputRow

syscall

li $v0,1

move $a0,$t8

syscall

li $v0,4

la $a0,colon

syscall

li $v0,4

la $a0,space

syscall

addi $t8, $t8, 1

jr $ra

#End subroutine loadArray

#subroutine to initializeVariables

initializeVariables:

li $t0, 0

li $t5, 4

li $t6, 0

li $t7, 4

li $t8, 1

jr $ra

#End subroutine initializeVariables