```
# CS240
# Lab 4 assignment – Solution
# compares a string entered by the user to the elements of an array of equal length strings, and prints
# a message if the string entered is a match for one of the elements in the array
            .data
length_validops: .word 5
length_op:         .word 3
validops:          .ascii "add","sub","and","slt","beq"
buffer:            .space 20          # string from keyboard stored here
maxlength:         .word 20
length_string:     .space 4           # length of string from keyboard stored here

prompt: .asciiz "Enter a string: "
result: .asciiz "\nValid operation "


                   .text
                   .globl main
main:              li $v0,4  # prompt for a string
                   la $a0,prompt
                   syscall

        # read in a string from the keyboard into address buffer in memory (max chars 20)
                   li $v0,8
                   la $a0,buffer
                   lw $a1,maxlength
                   syscall            # address of string in $a0

        # find length of string by looping and counting characters until a lf occurs
                   move $s0,$a0       # address of buffer in $s0
                   li $t0,'\n'
                   li $t1,0           # length of string in $t1
lengthloop:        lb $t2,0($s0)      # get a byte from the string
                   beq $t2,$t0,endlengthloop # if it is not a lf, add 1 to the count
                   addi $t1,$t1,1
                   addi $s0,$s0,1
                   j lengthloop       # and repeat
endlengthloop:   sw $t1,length_string # store the length of the string in memory

        # loop to check all the elements of the array against the string entered by the user
                   lw $t3,length_validops
                   lw $t4,length_op
                   la $a1,validops            # address of current element being examined in $a1
                                              # (initialized to base address)
checkloop:         beq $t3,$zero, exit
                   jal compareStrings         # compare the string with the current element of the array
                   bne $v0,$zero, endcheckloop      # if the result if 1, there is a match
                   addi $t3,$t3,-1             # if not a match, decrement the count of elements checked
                   add $a1,$a1,$t4
                   j checkloop                # and repeat
endcheckloop:    li $v0,4                     # print the success message is there is a match
                   la $a0,result
                   syscall

exit:              li $v0,10
                   syscall
```

```
# compareStrings takes 2 parameters, $a0 is the address of the string input by the user,
# $a1 is the address of the current element in  the array to compare the string to
# the return value in $v0 will be 1 if the string was a match, and 0 otherwise (1 = true, 0 = false)


compareStrings: move $s3,$a0
                move $s4,$a1
comparelength:  lw $t0,length_op          #compare lengths of the two strings
                lw $t1,length_string
                bne $t0,$t1,notfound       #if lengths are not equal, strings do not match

        # lengths are equal, so loop and compare the characters one by one of the two strings
                li $v0,1                   # initialize result to 1 (assume strings match, loop will set
                                           # result to 0 if there is not a match)
compareloop:    beq $t0,$zero,done
                lb $t5,0($s3)              # compare the two strings, character by character
                lb $t6,0($s4)
                bne $t5,$t6,notfound       # as soon as there's a mismatch, get out
                addi $s3,$s3,1
                addi $s4,$s4,1
                addi $t0,$t0,-1
                j compareloop              # repeat with the next pair of characters

notfound:       li $v0,0                   # set result to 0 (strings do not match)

done:           jr $ra
```