

```
.data
nums:    .word 8, 15, 2, 5, 13, 1, 22, 10, 17, 9, 7, 12, 4
size:    .word 13
sum:     .word 0
line:    .asciiz "\n"

.text

li $t0, 0    #used for counter/ i
li $t1, 10   #if conditional

#store inputs
la $s0, nums #store base adress of array
lw $s1, size
lw $s2, sum

#for loop
loop:      slt $t3, $t0, $s1 #if i < size return 1
           beq $t3, $zero, end #if size=i then step out of loop

           #shift array
           sll $t4, $t0, 2
           add $t4, $t4, $s0 #add shift to base address
           lw $t4, 0($t4) #load data at memory index

           #IF
           slt $t5, $t4, $t1 #if a[i]<10 return 1
           beq $t5, $zero, ifNot #if a[i]>10 skip if
           add $s3, $s3, $t4 #increase sum

           #print a[i]
           li $v0, 1 #used to print ints
           add $a0, $t4, $zero
           syscall
           li $v0, 4
           la $a0, line #insert new line
           syscall

ifNot:     li $v0, 1 #used to print ints
           add $a0, $s3, $zero #print sum
           syscall
           li $v0, 4
           la $a0, line #insert new line
           syscall

           addi $t0, $t0, 1 #increase for count (i++)
           j loop #jump to start

end:

#terminate program
li $v0, 10
syscall
```

MARS 4.4 Copyright 2003-2013 Pete Sanderson and Kenneth Vollmar

```
8
8
8
2
10
5
15
15
1
16
16
16
16
9
25
7
32
32
4
36

$s0      268500992
$s1      13
$s2      0
Mem[0x10010000] 8      15      2      5
Mem[0x10010010] 13     1      22     10
Mem[0x10010020] 17     9      7      12
Mem[0x10010030] 4      13     0
\n\nDiffs:\n\n
24c24
< $s0      268500992
---
> $s0      268501044
26c26
< $s2      0
---
> $s2      36
30c30
< Mem[0x10010030]      4      13      0
---
> Mem[0x10010030]      4      13      36
```