

Programma 1

```
.text
.globl main
main:  la      $t0,array    #carico indirizzo del vettore in T0
      li      $t1,3      #carico lunghezza del vettore in T1

      li      $v0,1      #imposto V0 per il successo
loop:  lw      $t2,0($t0)  #prendo un elemento del vettore...
      beqz    $t2,end    #...e se è 0 termino
      addi   $t0,4      #mi sposto sul prossimo elemento
      addi   $t1,-1     #aggiorno gli elementi rimanenti
      bgtz   $t1,loop   #se ho ancora elementi ricomincio
      li      $v0,-1    #imposto V0 per il fallimento

end:   move    $a0,$v0   #preparo la chiamata di stampa...
      li      $v0,1     #...del valore di V0
      syscall          #chiamata di sistema

.data
array: .word  -1,5,4    #un vettore di prova
```

Programma 2

```
main:  la      $t0,string #carico indirizzo della stringa in T0

      li      $v0,0      #inizializzo V0
loop:  lb      $t1,0($t0) #prendo un elemento della stringa
      beqz    $t1,end    #...e se è 0 termino
      sub     $t1,$t1,'0' #converto la cifra in numero
      mul    $v0,$v0,10  #moltiplico V0 per 10
      add    $v0,$v0,$t1 #e vi sommo la nuova cifra
      addi   $t0,1      #mi sposto sul prossimo elemento
      j      loop       #ricomincio

end:   move    $a0,$v0   #preparo la chiamata di stampa...
      li      $v0,1     #...del valore di V0
      syscall          #chiamata di sistema

.data
string: .asciiz "169634" #una stringa di prova
```

Programma 3

```
main:  la      $t0,array    #carico indirizzo del vettore in T0
      li      $t1,3      #carico lunghezza del vettore in T1

      li      $v0,0      #imposto V0 per il fallimento
loop:  lb      $t2,0($t0)  #prendo un elemento del vettore...
      bltz   $t2,end    #...e se è <0 termino
      bgt    $t2,16,end  #...e se è >16 termino
      addi   $t0,1      #mi sposto sul prossimo elemento
      addi   $t1,-1     #aggiorno gli elementi rimanenti
      bgtz   $t1,loop   #se ho ancora elementi ricomincio
      li      $v0,-1    #imposto V0 per il successo
```