

Programma 1

```
        li        $v0,0x80000000 #imposto V0 per il max (=min_int)
        li        $v1,0x7FFFFFFF #imposto V1 per il min (=max_int)
loop:   lw        $t2,0($t0)      #prendo un elemento del vettore
        ble      $t2,$v0,next1   #se è più piccolo di V0 continuo
        move     $v0,$t2        #altrimenti aggiorno V0
next1:  bge      $t2,$v1,next2   #se è più grande di V1 continuo
        move     $v1,$t2        #altrimenti aggiorno V1
next2:  addi     $t0,4           #mi sposto sul prossimo elemento
        addi     $t1,-1         #aggiorno gli elementi rimanenti
        bgtz    $t1,loop        #se ho ancora elementi ricomincio
```

Programma 2

```
main:   la        $t0,array      #t0=base address array
        li        $t1,7         #t2=lungh. array

        lb        $t2,0($t0)    #salvo primo elemento array
        j        next
loop:   lb        $t3,0($t0)    #in coppia con la prox istr..
        sb        $t3,-1($t0)   #copio indietro un byte
next:   addi     $t0,1          #mi sposto sul prossimo elemento
        addi     $t1,-1         #aggiorno gli elementi rimanenti
        bgtz    $t1,loop        #se ho ancora elementi ricomincio
        sb        $t2,-1($t0)   #memorizza primo elemento in fondo
```

Programma 3

```
main:   la        $t0,stringa   #t0=base address stringa

        li        $t3,'A'
        sub      $t3,$t3,'a'    #calcolo increm. minuscole->maiuscole

loop:   lb        $t1,0($t0)    #t1 contiene elemento stringa
        beqz    $t1,end        #se la stringa è vuota esce dal ciclo
        addi    $t0,$t0,1      #prossimo indirizzo da leggere
        bgt    $t1,'z',loop    #se maggiore di 'z' continuo ciclo
        blt    $t1,'a',loop    #se minore di 'a' continuo ciclo
        add    $t1,$t1,$t3     #cambia da minuscola a maiuscola
        sb     $t1,-1($t0)    #memorizza la nuova lettera
        j loop #continua ciclo
end:   ...
```

Programma 4

```
main:   li        $a0,3

        li        $v0,1        #inizializza accumulatore
loop:   blt      $a0,2,end      #se t0 < 2 esci
        mul      $v0,$v0,$a0    #v0 = v0 * a0
        addi    $a0,-1         #decrementa contatore
        j        loop          #continua ciclo
end:   ...
```