

```

modifica:  addi   $sp,-12      #diminuisce di 12 byte lo stack pointer
           sw     $ra,0($sp)  #salva sullo stack il registro $ra
           sw     $s0,4($sp)  #salva sullo stack il registro $s0
           sw     $s1,8($sp)  #salva sullo stack il registro $s1
           move   $s0,$a0     #sposta in s0 l'indirizzo base dell'array
           lw     $s1,0($s0)  #carica in s1 la lunghezza dell'array

loop:      lw     $a0,4($s0)   #carica in a0 l'elemento corrente dell'array
           bltz  $a0,next     #se a0 < 0, salta a next
           jal   fatt         #chiama la subroutine fatt
           sw     $v0,4($s0)  #salva il fattoriale nell' array

next:      addi   $s1,-1      #diminuisce di 1 la lunghezza dell'array da visitare
           addi   $s0,4      #aumenta di 4 byte l'indirizzo base dell'array
           bgtz  $s1,loop    #se s1 > 0, salta a loop
           lw     $ra,0($sp)  #carica dallo stack il registro $ra
           lw     $s0,4($sp)  #carica dallo stack il registro $s0
           lw     $s1,8($sp)  #carica dallo stack il registro $s1
           addi   $sp,12
           jr    $ra

ver_mat:   li     $v0,-1     #imposto inizialmente v0 per matrice negativa

loopRow:   move   $t0,$a2     #copia in t0 il numero di colonne
           move   $t2,$a0     #copia in t2 l'address base della matrice

loop:      lb     $t1,0($t2)  #carica in t1 l'elemento corrente
           bltz  $t1,nextRow  #se t1 < 0, passa alla prossima riga
           addi   $t2,1      #altrimenti aumenta di 1 byte l'indirizzo base della matrice
           addi   $t0,-1     #diminuisce di 1 il numero di colonne da visitare
           bgtz  $t0,loop    #se t0 > 0, salta a "loop"
           li    $v0,1      #altrimenti imposto v0 per matrice positiva
           j     end        #salto incondizionato a "end"

nextRow:   addi   $a1,-1     #diminuisce di 1 il numero di righe da visitare
           addi   $a0,4      #sposta l'address base alla riga successiva
           bgtz  $a1,loopRow #se a1 > 0, salta a "loopRow"

end:      jr    $ra

```