```
# EXERCICIO 2 - FATORIAL

                  .data
                  .align 0
dig_txt:          .asciiz "Digite um numero para o calculo fatorial: "
res_txt:          .asciiz "O resultado eh: "
                  .text
                  .globl main
main:# 1. Lendo um numero ---------------------------------------------------
      li    $v0,  4                 # print string code
      la    $a0,  dig_txt           # load dig_txt
      syscall                       # system, do it!

      li    $v0,  5                 # read int code
      syscall                       # system, do it!
      move  $s0,  $v0               # $s0 = input num

      # 2. Iniciando a operação de fatorial ---------------------------------
      jal factorial                 # do factorial
      move  $s1,  $v0               # $s1 = result

      # 3. Imprimindo o resultado -------------------------------------------
      li    $v0,  4                 # print string code
      la    $a0,  res_txt           # load res_txt
      syscall                       # system, do it!

      li    $v0,  1                 # print int code
      move  $a0,  $s1               # load result
      syscall                       # system, do it!

      # 4. Encerrando o programa...
      li    $v0,  10                # exit code
      syscall                       # system, do it!

#:::::::::::::::::::: FACTORIAL ::::::::::::::::::::::
factorial:
      # 1. criando a pilha da chamada da funcao
      addi  $sp,  $sp,  -8          # adjusting stack
      sw    $s0,  4($sp)            # save start num
      sw    $ra,  0($sp)            # save return address

      li    $v0,  1     # fat = 1
      lw    $t0,  4($sp)            # aux = N (numero inicial)

      # 2.while N > 0 do calculo_do_fatorial
fat_loop:
      ble   $t0,  0,    end_floop   # if $t0 <= 0, end loop
      mul   $v0,  $v0,  $t0         # fat = fat * aux
      sub   $t0,  $t0,  1           # N--
      j fat_loop                    # while $t0 > 0, goto loop
end_floop:

      # 3.liberando a pilha e encerrando a funcao
      lw    $s0,  4($sp)            # load start num
      lw    $ra,  0($sp)            # load return address
      sub   $sp,  $sp,  -8          # readjusting stack

      jr    $ra                     #return to call func
```