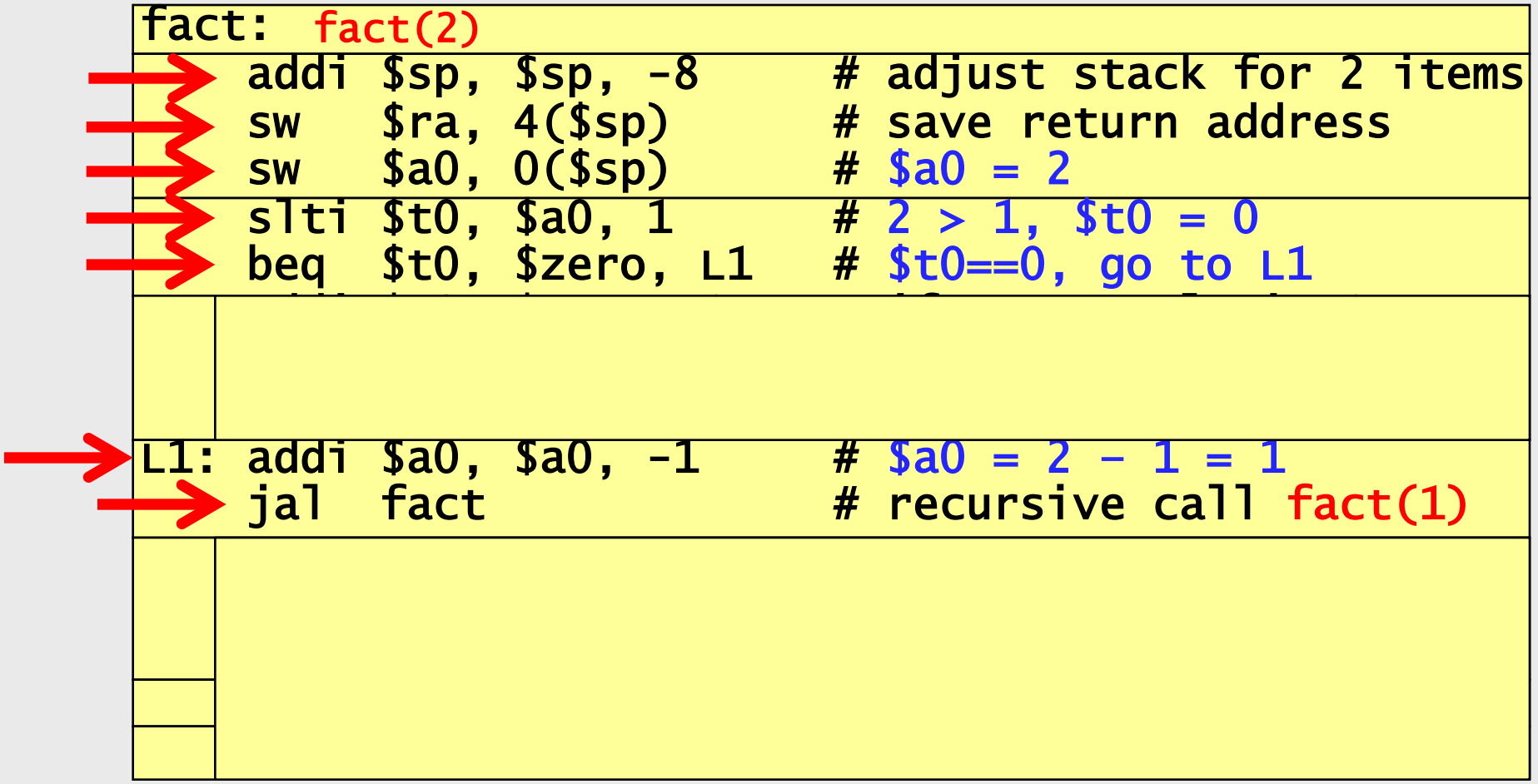
Non-Leaf Procedure Example

- MIPS code:

```
fact:
    addi $sp, $sp, -8       # adjust stack for 2 items
    sw   $ra, 4($sp)        # save return address
    sw   $a0, 0($sp)        # save argument
    slti $t0, $a0, 1        # test for n < 1
    beq  $t0, $zero, L1
    addi $v0, $zero, 1      # if so, result is 1
    addi $sp, $sp, 8        #   pop 2 items from stack
    jr   $ra                #   and return
L1: addi $a0, $a0, -1       # else decrement n
    jal  fact               # recursive call
    lw   $a0, 0($sp)        # restore original n
    lw   $ra, 4($sp)        #   and return address
    addi $sp, $sp, 8        # pop 2 items from stack
    mul  $v0, $a0, $v0      # multiply to get result
    jr   $ra                # and return
```

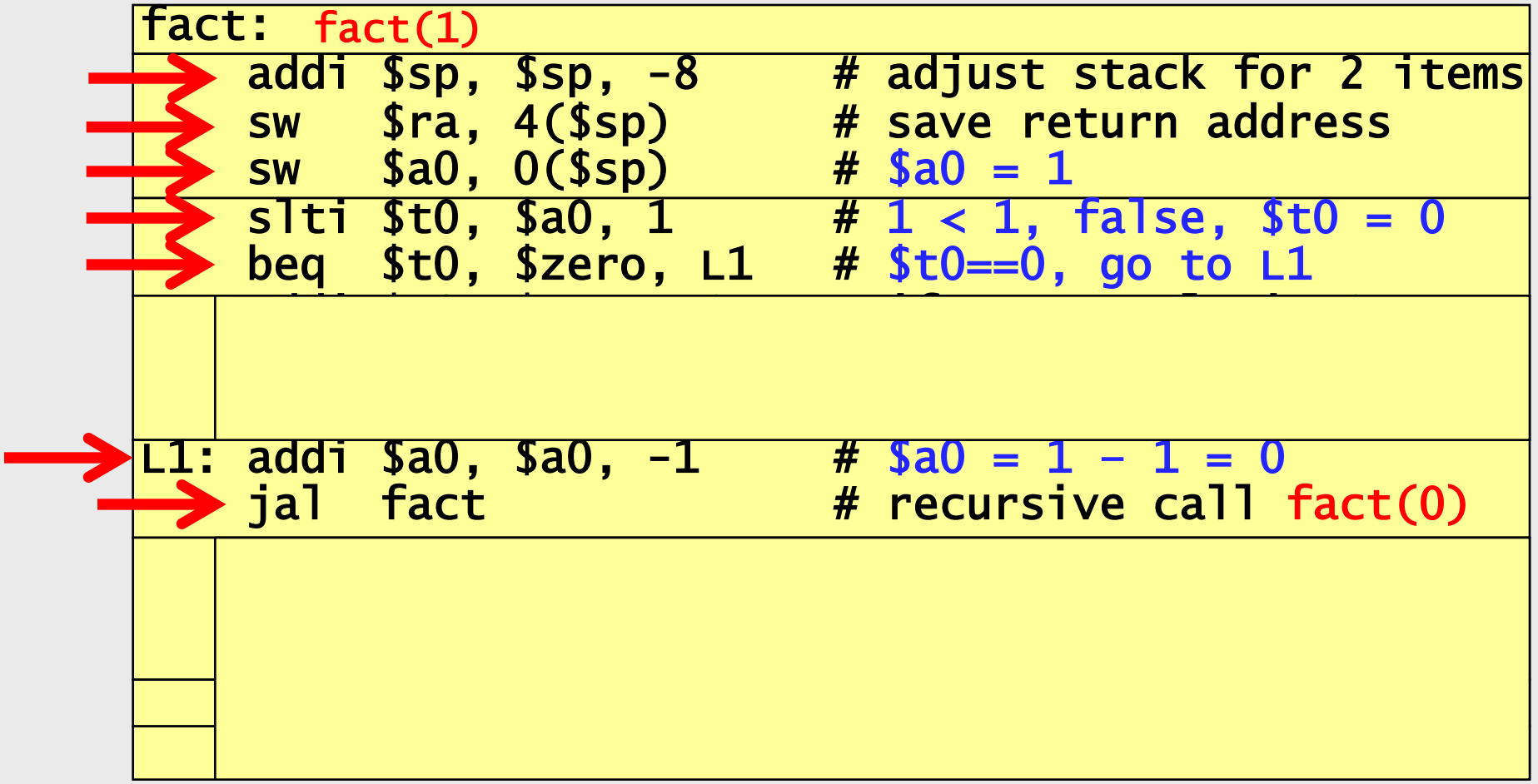
Non-Leaf Procedure Example

- MIPS code: (example of fact (2))



Non-Leaf Procedure Example

- MIPS code: (example of fact (2))



Non-Leaf Procedure Example

- MIPS code: (example of fact (2))

```
fact:  fact(0)
        addi $sp, $sp, -8       # adjust stack for 2 items
        sw   $ra, 4($sp)        # save return address
        sw   $a0, 0($sp)        # $a0 = 0
        slti $t0, $a0, 1        # 0 < 1, $t0 = 1
        beq  $t0, $zero, L1     # $t0!=0, go on
        addi $v0, $zero, 1      # if so, result is 1
        addi $sp, $sp, 8        #   pop 2 items from stack
        jr   $ra                #   and return to fact(1)
L1:
```

Non-Leaf Procedure Example

- MIPS code: (example of fact (2))

```
fact:  fact(1)
       addi $sp, $sp, -8       # adjust stack for 2 items
       sw   $ra, 4($sp)        # save return address
       sw   $a0, 0($sp)        # $a0 = 1
       slti $t0, $a0, 1        # 1 < 1, false, $t0 = 0
       beq  $t0, $zero, L1     # $t0==0, go to L1
       
L1:    addi $a0, $a0, -1       # $a0 = 1 - 1 = 0
       jal  fact               # recursive call fact(0)
XYZ    lw   $a0, 0($sp)        # restore original n (1)
       lw   $ra, 4($sp)        #   and return address
       addi $sp, $sp, 8        # pop 2 items from stack
       mul  $v0, $a0, $v0      # $v0 = 1 x 1 = 1
       jr   $ra                # and return to fact(2)
```

Non-Leaf Procedure Example

- MIPS code: (example of fact (2))

```
fact:  fact(2)
    addi $sp, $sp, -8       # adjust stack for 2 items
    sw   $ra, 4($sp)        # save return address
    sw   $a0, 0($sp)        # $a0 = 2
    slti $t0, $a0, 1        # 2 > 1, $t0 = 0
    beq  $t0, $zero, L1     # $t0==0, go to L1
    
L1: addi $a0, $a0, -1       # $a0 = 2 - 1 = 1
    jal  fact               # recursive call fact(1)
    lw   $a0, 0($sp)        # restore original n(2)
    lw   $ra, 4($sp)        #   and return address
    addi $sp, $sp, 8        # pop 2 items from stack
    mul  $v0, $a0, $v0      # $v0 = 2 x 1 = 2
    jr   $ra                # and return
```

XYZ

fact(2) = 2

Non-Leaf Procedure Example
Local Data on the Stack