

# Assembly RISC-V: ricorsione, compiler e linker

**Autore Principale**

# Esempio: funzione ricorsiva

Consideriamo un programma ricorsivo per calcolare il fattoriale

```
#include<iostream>
using namespace std;

int fattoriale(int n){
    int f;
    if(n==1) return 1;
    f=fattoriale(n-1)*n;
    return f;
}
main(){
    cout<<fattoriale(5);
}
```

# Codice assembly

```
.globl main
.data
m1: .string "! = "
m2: .string " \n"

.macro exit (%x)
    li a7, 93
    li a0, %x
    ecall
.end_macro

.macro print_str (%x)
    li a7, 4
    la a0, %x
    ecall
.end_macro

.macro print_int (%x)
    li a7, 1
    mv a0, %x
    ecall
.end_macro

.text
main:
    li t3, -1 # per cancellare contenuto stack
    li t4, 1  # usato per controllo caso base
    li s0, 5  # salvo argomento
    mv a0, s0 # preparo argomento della chiamata
    jal fattoriale # chiamo la funzione
    mv s1, a0 # salvo il risultato
    print_int(s0) #stampo !s0=s1
    print_str(m1)
    print_int(s1)
    print_str(m2)
    exit(0)

fattoriale:
    addi sp, sp, -8 # alloco call-frame
    sw a0, 4(sp) # salvo argomento x sullo stack (servirà per x*fatt(x-1)
    sw ra, 8(sp) # salvo indirizzo di ritorno sullo stack
test:
    beq a0, t4, base # se a0 == 1 salto a "base"
passo:
    addi a0, a0, -1 # preparo parametro per la chiamata ricorsiva in a0
    jal fattoriale # chiamo fattoriale(x-1)
    mv t1, a0 # salvo risultato di fatt(x-1)
    lw a0, 4(sp) # ripristino parametro in a0
    lw ra, 8(sp) # ripristino indirizzo di ritorno in ra
    sw t3, 4(sp) # reset del call frame
    sw t3, 8(sp) # " "
    mul a0, a0, t1 # calcolo x*fatt(x-1)
base:
    addi sp, sp, 8 # dealloco call-frame
    ret
```

# Inserisco breakpoint: chiamata ricorsiva e indirizzo di ritorno

Text Segment				
Bkpt	Address	Code	Basic	Source
<input type="checkbox"/>	0x00400058	0x00000073	ecall	<9> ecall
<input type="checkbox"/>	0x0040005c	0xff810113	addi x2,x2,-8	39: addi sp, sp, -8 # alloco call-frame
<input type="checkbox"/>	0x00400060	0x00a12223	sw x10,4(x2)	40: sw a0, 4(sp) # salvo argomento x sullo st...
<input type="checkbox"/>	0x00400064	0x00112423	sw x1,8(x2)	41: sw ra, 8(sp) # salvo indirizzo di ritorno...
<input type="checkbox"/>	0x00400068	0x03d50263	beq x10,x29,18	43: beq a0, t4, base # se a0 == 1 salto a "base"
<input type="checkbox"/>	0x0040006c	0xffff50513	addi x10,x10,-1	45: addi a0, a0, -1 # preparo parametro per la ...
<input checked="" type="checkbox"/>	0x00400070	0xfedff0ef	jal x1,-10	46: jal fattoriale # chiamo fattoriale(x-1)
<input checked="" type="checkbox"/>	0x00400074	0x00a00333	add x6,x0,x10	47: mv t1, a0 # salvo risultato di fatt(x-1)
<input type="checkbox"/>	0x00400078	0x00412503	lw x10,4(x2)	48: lw a0, 4(sp) # ripristino parametro in a0
<input type="checkbox"/>	0x0040007c	0x00812083	lw x1,8(x2)	49: lw ra, 8(sp) # ripristino indirizzo di r...
<input type="checkbox"/>	0x00400080	0x01c12223	sw x28,4(x2)	50: sw t3, 4(sp) # reset del call frame
<input type="checkbox"/>	0x00400084	0x01c12423	sw x28,8(x2)	51: sw t3, 8(sp) # " "
<input type="checkbox"/>	0x00400088	0x02650533	mul x10,x10,x6	52: mul a0,a0,t1 # calcolo x*fatt(x-1)
<input type="checkbox"/>	0x0040008c	0x00810113	addi x2,x2,8	54: addi sp, sp, 8 # dealloco call-frame
<input type="checkbox"/>	0x00400090	0x00008067	jalr x0,x1,0	55: ret

Chiamata ricorsiva: fattoriale(n-1)

Indirizzo di ritorno (istruzione dopo la chiamata ricorsiva)

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x7ffffe0	0	0	0	0	0	0	0	5 4194324
0x7ffff00	0	0	0	0	0	0	0	0 0
0x7ffff020	0	0	0	0	0	0	0	0 0
0x7ffff040	0	0	0	0	0	0	0	0 0
0x7ffff060	0	0	0	0	0	0	0	0 0
0x7ffff080	0	0	0	0	0	0	0	0 0
0x7ffff0a0	0	0	0	0	0	0	0	0 0
0x7ffff0c0	0	0	0	0	0	0	0	0 0
0x7ffff0e0	0	0	0	0	0	0	0	0 0
0x7ffff100	0	0	0	0	0	0	0	0 0
0x7ffff120	0	0	0	0	0	0	0	0 0
0x7ffff140	0	0	0	0	0	0	0	0 0
0x7ffff160	0	0	0	0	0	0	0	0 0

fattoriale(5)  
Indirizzo di ritorno  
(nel main)  
argomento=5 salvato

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x7ffffe0	0	0	0	3 4194420	4 4194420	5 4194324		
0x7ffff00	0	0	0	0	0	0	0	0
0x7ffff020	0	0	0	0	0	0	0	0
0x7ffff040	0	0	0	0	0	0	0	0
0x7ffff060	0	0	0	0	0	0	0	0
0x7ffff080	0	0	0	0	0	0	0	0
0x7ffff0a0	0	0	0	0	0	0	0	0
0x7ffff0c0	0	0	0	0	0	0	0	0
0x7ffff0e0	0	0	0	0	0	0	0	0
0x7ffff100	0	0	0	0	0	0	0	0
0x7ffff120	0	0	0	0	0	0	0	0
0x7ffff140	0	0	0	0	0	0	0	0
0x7ffff160	0	0	0	0	0	0	0	0

stesso indirizzo!

Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)
0x7ffffe0	0	0	0	3 4194420	4 4194420	5 4194324		
0x7ffff00	0	0	0	0	0	0	0	0
0x7ffff020	0	0	0	0	0	0	0	0
0x7ffff040	0	0	0	0	0	0	0	0
0x7ffff060	0	0	0	0	0	0	0	0
0x7ffff080	0	0	0	0	0	0	0	0
0x7ffff0a0	0	0	0	0	0	0	0	0
0x7ffff0c0	0	0	0	0	0	0	0	0
0x7ffff0e0	0	0	0	0	0	0	0	0
0x7ffff100	0	0	0	0	0	0	0	0
0x7ffff120	0	0	0	0	0	0	0	0
0x7ffff140	0	0	0	0	0	0	0	0
0x7ffff160	0	0	0	0	0	0	0	0

fattoriale(4)  
argomento=4  
indirizzo di ritorno (in fattoriale)  
  
fattoriale(3)  
argomento=(3)  
indirizzo di ritorno (in fattoriale)

Data Segment									
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)	Value (+20)
0x7ffffefe0	-1	-1	-1	-1	-1	4	4194420	5	4194324
0x7fffff000	0	0	0	0	0	0	0	0	0
0x7fffff020	0	0	0	0	0	0	0	0	0
0x7fffff040	0	0	0	0	0	0	0	0	0
0x7fffff060	0	0	0	0	0	0	0	0	0
0x7fffff080	0	0	0	0	0	0	0	0	0
0x7fffff0a0	0	0	0	0	0	0	0	0	0
0x7fffff0c0	0	0	0	0	0	0	0	0	0
0x7fffff0e0	0	0	0	0	0	0	0	0	0
0x7fffff100	0	0	0	0	0	0	0	0	0
0x7fffff120	0	0	0	0	0	0	0	0	0
0x7fffff140	0	0	0	0	0	0	0	0	0
0x7fffff160	0	0	0	0	0	0	0	0	0



current sp



☒ Hexadecimal Addresses

☐ Hexadecimal Values

☐ ASCII

Data Segment									
Address	Value (+0)	Value (+4)	Value (+8)	Value (+c)	Value (+10)	Value (+14)	Value (+18)	Value (+1c)	Value (+20)
0x7ffffefe0	-1	-1	-1	-1	-1	-1	-1	5	4194324
0x7fffff000	0	0	0	0	0	0	0	0	0
0x7fffff020	0	0	0	0	0	0	0	0	0
0x7fffff040	0	0	0	0	0	0	0	0	0
0x7fffff060	0	0	0	0	0	0	0	0	0
0x7fffff080	0	0	0	0	0	0	0	0	0
0x7fffff0a0	0	0	0	0	0	0	0	0	0
0x7fffff0c0	0	0	0	0	0	0	0	0	0
0x7fffff0e0	0	0	0	0	0	0	0	0	0
0x7fffff100	0	0	0	0	0	0	0	0	0
0x7fffff120	0	0	0	0	0	0	0	0	0
0x7fffff140	0	0	0	0	0	0	0	0	0
0x7fffff160	0	0	0	0	0	0	0	0	0



current sp



☒ Hexadecimal Addresses

☐ Hexadecimal Values

☐ ASCII

Ritorno dalle chiamate ricorsive: lo stack decresce!  
(-1 = record cancellato)