

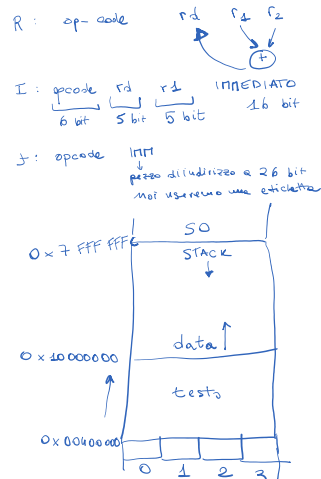
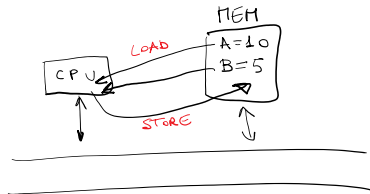
Esercitazione 1 sq1

29 September 2020 13:31



01.ESERCITAZIONI_A...

Traduzione da C ad assembly



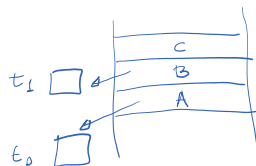
DIRETTIVE

- **WORD** 1 sta allocando 32 bit (4 byte) mettendo il valore 1
- **SPACE** N

intero 4 byte

```
int a = 1;
int b = 2;
int c = 3;
int main ()
{
    if( a==b && a == c)
        a++;
    else
        b++;
    c = 12;
}
```

```
.data
A: .word 1
B: .word 2
C: .word 3
.text
MAIN:
lw $t0, A
lw $t1, B
bne $t0, $t1, ELSE
lw $t1, C
bne $t0, $t1, ELSE
```

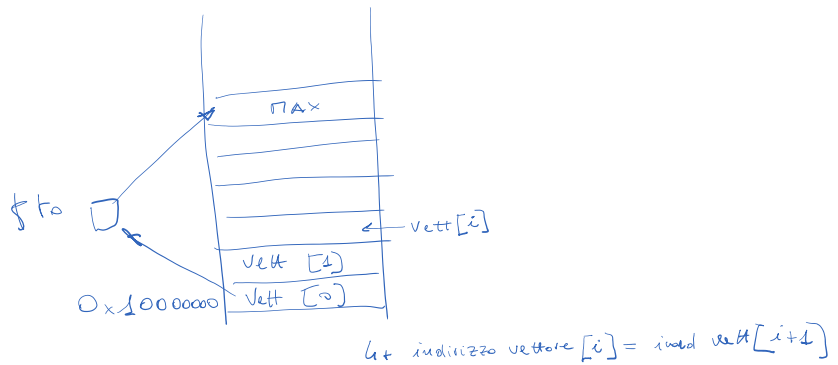


```
lw $t0, A
addi $t0, $t0, 1
sw $t0, A
j ENDIF
```

```
ELSE:
lw $t0, B($gp)
addiu $t0, $t0, 1
sw $t0, B($gp)
ENDIF:
li $t0, 12
sw $t0, C
```

```
li $v0, 10
syscall
```

altri esercizi vedasi nel pdf linkato



$\star = 32 \text{ bit}$
(sono indirizzi)
vettore di puntatori

21V N13
40 N13

0x10000010	PTR
0x10000008	PTRLIST[3]
0x10000004	PTRLIST[2]
0x10000000	PTRLIST[0]
0x10000000	VAL

VAL 5

La funzione chiamante salva i suoi registri temporanei se sono pieni e se devono essere utilizzati dopo.
Poi salva i registri degli argomenti
ogni funzione ha degli argomenti

FUNZ(a,b) → \$a0 = a
\$a1 = b
FUNZ(c) → **MA LO SOVRASCRIVE!**

non abbiamo nulla da salvare perché il main non ha argomenti.
Poi in v0 e v1 vengono salvati i valori di ritorno, alla fine il main si ritrova il valore output della funzione in v0.

Bisogna salvare i registri di ritorno se sono usati da altre sottofunzione

\$a0 = 3
\$a1 = val

li \$a0, 3
lw \$a1, VAL
jal PTRLOOP
sw \$v0, PR

ANDATA E RITORNO
PC & PTRLOOP
nel \$ra si salva l'attuale PC

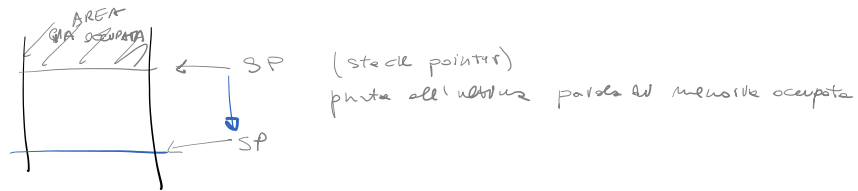
PR
SI TROVERA' IN \$v0 il risultato di PTRLOOP

Epilogo del chiamante, ma questo è un esempio dove non c'è bisogno

Area di attivazione segmento, per ogni
funzione, dello stack che cresce verso il
basso
In questo segmento vengono salvati
valori ..., variabili locali, backup vari in
modo da poter sovrascrivere.

AREA
OCCUPATA

modo da poter sovrascrivere.



prologo del chiamato

- se necessario salvare fp (non serve qui)
- se funzione foglia, funzione che non ne chiama altre,...ma la nostra non lo è occorre quindi salvare il return address (ra)
- se necessario salvare i registri s (quelli da s0 a s7)

