



Assembly

Formato R

- primi 6 bit: **codice operativo**
- 3 celle da 5 bit: **registri degli operatori**
- 5 bit di **shift amount**
- ultimi 6 bit: **tipo di operatore**

codice operativo	registro operatore	registro operatore	registro operatore	shift amount	tipo di operatore
------------------	--------------------	--------------------	--------------------	--------------	-------------------

Formato I

- primi 6 bit: **codice operativo, nelle istruzioni I vale 8**
- 1 cella da 5 bit: **registri di destinazione**
- 1 cella da 5 bit per il **registro target**
- ultimi 16 bit: **operando in CA2**

codice operativo	registro di destinazione	registro target	operando in CA2
------------------	--------------------------	-----------------	-----------------

Per esempio `i++`

Alterazione del flusso

BNE *branch on equal*

Salto relativo o condizionato

- primi 6 bit: **codice operativo, nelle istruzioni I vale 8**
- 1 cella da 5 bit: **registri di destinazione**
- 1 cella da 5 bit per il **registro target**
- ultimi 16 bit: **indirizzo dell'istruzione**

codice operativo	registro di destinazione	registro target	indirizzo dell'istruzione
------------------	--------------------------	-----------------	---------------------------

JUMP, formato J

- 1 cella da 6 bit per il **codice operativo**
- ultimi 26 bit: **Registro di destinazione**

codice operativo	registro di destinazione
------------------	--------------------------

Esiste anche la funzione Jump and link che oltre che effettuare il salto salva nel registro \$ra il punto di partenza.

Gestione della memoria

La memoria del pc è suddivisa in due parti, una riservata nella quale i programmi in esecuzione non possono sovrascrivere nulla e una parte invece disponibile ai programmi dell'utente, la parte disponibile all'utente è suddivisa così:

Stack
Heap
Static
Text

Lo stack e l'Heap fanno parte della stessa sezione di memoria ma si sviluppano in due sensi opposti, lo stack dall'alto verso il basso e l'heap dal basso verso l'alto.

Registri temporanei \$t

Si tratta di registri che sono accessibili dai metodi dei programmi i quali possono essere modificati a piacere.

Registri saved temporanei \$s

Si tratta di registri accessibili dai metodi dei programmi i quali possono essere modificati a piacimento ma una volta conclusa l'esecuzione del metodo devono essere reimpostati al valore precedente alla chiamata.

Registri per le procedure

Le procedure utilizzano i registri **da \$a0 fino a \$a3** per salvare i loro **parametri** mentre utilizzano **\$v0 e \$v1** per salvare i **valori in uscita**

Syscall

Il **syscall** simula alcune funzioni basilari del sistema operativo come per esempio la print.

Per richiamare un servizio syscall è necessario assegnare a \$v0 il codice corrispondente e poi inserire in \$a0...\$a3 i parametri necessari, se la procedura prevede un valore di ritorno esso sarà salvato in \$v0 o \$a0.