

## I registri in RISC-V

In RISC-V, abbiamo **32 registri a 64 bit** (chiamati x0 a x31), che possiamo usare per diversi scopi. Alcuni di questi registri hanno ruoli specifici, mentre altri sono liberi per uso generale.

Ecco una breve panoramica dei tipi di registri principali:

### 1. Registri temporanei (t0 - t6, x5 - x7, x28 - x31):

- Usati per **memorizzare dati temporanei**.
- Non devono essere preservati tra le chiamate di funzione (ovvero, se viene chiamata una funzione, il contenuto può essere sovrascritto senza bisogno di ripristinarlo).
- Sono i registri ideali quando si fanno operazioni intermedie, come somme, carichi e salvataggi in memoria.

### 2. Registri salvati (s0 - s11, x8 - x18):

- Devono essere **preservati** attraverso le chiamate di funzione (ovvero, una funzione che usa questi registri deve ripristinare il loro valore originale prima di restituire il controllo).
- Usati per mantenere valori importanti che devono essere conservati durante tutto il programma o attraverso le chiamate di funzione.

### 3. Registri argomento (a0 - a7, x10 - x17):

- Utilizzati per **passare argomenti e restituire valori** nelle chiamate di funzione.

### 4. Registro zero (x0):

- Sempre uguale a 0, utile per fare operazioni che coinvolgono lo zero.

---

## Registri temporanei: Come funzionano

I **registri temporanei** (t0 - t6, x5 - x7, x28 - x31) vengono usati quando abbiamo bisogno di registri **"usa e getta"**, ovvero registri che non richiedono di mantenere il valore tra le diverse operazioni o funzioni.

Nell'esempio che stiamo vedendo, x9 è usato proprio come registro temporaneo, anche se x9 non è tipicamente un registro "temporaneo" predefinito, è libero da vincoli particolari in questo contesto.

---

## Perché usare i registri temporanei?

Quando fai operazioni su dati in memoria (come nel nostro caso con l'array *A*), ci sono diversi passaggi intermedi che necessitano di un registro temporaneo per memorizzare i dati. Ecco cosa succede:

1. **Accedere ai dati in memoria:** In RISC-V, i dati devono essere **caricati dalla memoria in un registro** per poter essere manipolati. Non possiamo sommare direttamente valori in memoria, ma dobbiamo prima trasferirli in un registro.

- Nel nostro caso, abbiamo caricato il valore di  $A[8]$  in  $x9$  (registro temporaneo) con l'istruzione ``ld x9, 64(x22)``.

2. **Eseguire l'operazione:** Una volta che il dato è nel registro, possiamo eseguire l'operazione (in questo caso, sommare  $h$  a  $A[8]$ ).

- La somma  $h + A[8]$  avviene in  $x9$  stesso: ``add x9, x21, x9``.

3. **Salvare il risultato:** Dopo aver eseguito l'operazione, dobbiamo **salvare il risultato di nuovo in memoria**, nel nostro caso in  $A[12]$ .

- Usiamo ancora  $x9$  per contenere il risultato della somma, e poi lo scriviamo in  $A[12]$  con l'istruzione ``sd x9, 96(x22)``.
- 

## Uso dei registri temporanei in altri casi

Oltre al nostro esempio, i registri temporanei sono utili in vari contesti, come:

- **Operazioni intermedie nei cicli (loop):** Quando manipoli indici o fai conti ripetuti, puoi usare registri temporanei per mantenere i risultati parziali.
  - **Salvataggio di valori intermedi in calcoli complessi:** Se stai facendo un'operazione complessa (ad esempio, una moltiplicazione o una sequenza di operazioni), hai bisogno di registri temporanei per memorizzare i risultati intermedi.
  - **Accesso a valori in memoria:** Ogni volta che devi lavorare con i valori della memoria (come array o variabili globali), devi usare registri temporanei per caricare i dati, manipolarli e poi scrivere i risultati.
- 

## In sintesi:

1. **Carico in registro:** Quando lavoriamo con dati dalla memoria, li carichiamo in un registro temporaneo, poiché RISC-V non può operare direttamente sui dati in memoria.

2. **Esegui operazioni:** Usiamo i registri temporanei per eseguire operazioni intermedie (somma, sottrazione, ecc.).

3. **Salvo il risultato:** Una volta ottenuto il risultato, possiamo salvarlo di nuovo in memoria o usarlo per altre operazioni.

Nel caso del nostro esercizio, ``x9`` è stato scelto come registro temporaneo per immagazzinare il valore intermedio di  $A[8]$  e poi  $h + A[8]$ , prima di scrivere il risultato in memoria.