

# Compito A (tempo a disposizione: 2 ore)

---

## ■ DOMANDA 1 (15 minuti)

Con riferimento alla implementazione *multiciclo* della CPU della macchina MIPS senza pipelining, definire il concetto di *microprogramma* e illustrare e commentare lo schema di massima dell'unità di controllo

## ■ DOMANDA 2 (15 minuti)

Si consideri una memoria gerarchica comprendente un livello di cache e la memoria principale, con le seguenti caratteristiche:

- indirizzo da 16 bit
- blocchi da 4 parole
- cache ad indirizzamento diretto da 16 KB

Ipotizzando che la CPU richieda la parola di indirizzo

1 0 1 0 0 0 1 0 1 1 0 1 0 1 0 0

si mostri come viene utilizzato l'indirizzo per gestire gli accessi in cache (fornire lo schema della cache).

---

- **DOMANDA 3** (30 minuti)

Fornire una formulazione dettagliata del tempo di CPU relativo all'utente in una macchina con un livello di cache, **discutendo in particolare gli aspetti architetturali che incidono sul valore delle varie componenti e le interdipendenze tra tali componenti.**

- **DOMANDA 4** (20 minuti)

Con riferimento alla *pipeline con schedulazione dinamica* specificare brevemente ma senza ambiguità:

1. come vengono realizzate, in linea di principio, le unità funzionali
2. quale è il ruolo della stazione di consegna
3. che differenze ci sono tra il PowerPC e il PentiumPro relativamente al trattamento dei bit prelevati dalla cache istruzioni

## Domanda 5 (40 minuti)

- **Valutare il numero di cicli di clock necessari a completare l'esecuzione di questo programma nel caso di pipeline semplice (si ipotizzi che il ciclo si esegue 50 volte).**
  - Si supponga di essere nel caso ideale di assenza di fallimenti in cache (*specificare perché è necessaria questa precisazione*)
  - Si adotti la tecnica basata sulla predizione di fallimento per la gestione delle criticità sul controllo (*SPIEGARLA*)
  - Si adotti la tecnica di propagazione per la risoluzione delle criticità sui dati (*SPIEGARLA*)
  - Si supponga di non poter effettuare il riordinamento del codice per criticità di tipo carica-e-usa (*SPIEGARLO*)

```
add $t1,$t3,$s0
add $t0,$zero,$s0
loop: slt $t2,$t0,$t1
      beq $t2,$zero,exit
      lw  $t3,0($t0)
      add $s1,$s1,$t3
      addi $t0,$t0,4
      j  loop
exit: sw $s1,0($s4)
```

- ***N.B. Evidenziare tutti gli anticipi e gli stalli necessari, specificando il numero di volte per cui si ripetono***