

Cognome e nome dello studente:

Matricola:

1. [2] Modificare la CPU in Figura, in modo che possa gestire gli hazard sui dati, quando è in esecuzione il seguente segmento di codice:

```
0x0000 040C    lw $t1, 32($s0)
0x0000 0410    sw $t1, 64($s0)
```

2. [2] Modificare la CPU in Figura, in modo che possa gestire gli hazard sui dati anche nel caso di una istruzione di `beq`, ad esempio:

```
0x0000 0400    add $t1, $s0, $s1
0x0000 0404    beq $t1, $t0, 4
```

3. [5] Cosa sono gli interrupt e le eccezioni? Come vengono gestiti dalle architetture Intel e dalle architetture MIPS/ARM? Specificare gli elementi della CPU MIPS che sono dedicati alla gestione delle eccezioni e cosa contengono. Modificare la CPU in Figura per potere gestire un'eccezione di "Overflow". Cosa si intende per mascheramento degli interrupt? Viene praticato nei MIPS? Come vengono gestite le eccezioni e gli interrupt dai sistemi operativi sul MIPS?

4. [4] Quali sono gli elementi caratteristici del modulo di Fetch&Decode di una pipeline superscalare. Cosa sono le reservation station e la commit unit? A cosa servono? Che differenza c'è tra pipeline super-scalare e pipeline dotata di VLIW? Quali sono i vantaggi e gli svantaggi di un approccio rispetto all'altro. Qual è il migliore e perché?

5. [2] Come si implementa l'esecuzione vettoriale in una pipeline super-scalare? Mostrarlo modificando un cammino di esecuzione della pipeline di Figura 1, non dotata di esecuzione vettoriale.

6. [1] Spiegare chiaramente cosa si intende per stallo e illustrare almeno una tecnica per ottenerlo.

7. [5] Disegnare una memoria cache (parte dati + TAG + bit di validità) e la sua porta di lettura e scrittura. Utilizzare componenti three-state per minimizzare il numero di porte logiche. Considerate un'architettura MIPS a 64 bit, a 4 vie di 64 KByte per banco, e linee di 8 parole (per ciascun banco). Definire cosa rappresenta il campo TAG e dimensionarlo opportunamente. Dove posso trovare il dato letto dall'istruzione `lw $t1, 2048($0)`? Da quanti bit è costituita questa memoria complessivamente? Cosa succede quando si verifica una miss? Come si può limitare la frequenza di miss? Da quale dei quattro banchi viene scaricato il dato quando occorre caricare una nuova linea? Perché?

8. [3] Cosa sono i codici di rilevamento e correzione degli errori? Come funziona il codice di Hamming? Calcolare il codice per il numero binario 1000 0000. Chi utilizza il codice di rilevamento e correzione degli errori? Schizzare uno schema a blocchi di un modulo, che mediante il codice di Hamming, possa correggere un errore singolo.

9. [2] Che cos'è e che cosa rappresenta il "roof model"? Cosa rappresenta l'intensità aritmetica? Si riferisce ad una CPU o ad un particolare programma? Un programma che elabora matrici sparse sarà un programma con intensità aritmetica alta o bassa? Perché? Quali sono i passi per ottimizzare le prestazioni del codice suggeriti dal roof-model? Cos'è un kernel benchmark? Cos'è lo SPEC?

10. [2] Riportare alcune caratteristiche della architettura x86.

11. [3] Come viene gestito l'input/output dalle architetture MIPS e dalle architetture Intel? Identificare i componenti principali di un'interfaccia di una periferica. Descrivere il protocollo e la struttura di un collegamento daisy chain e di un arbitraggio centralizzato con priorità. Cosa sono i bridge? Come sono organizzati i dischi magnetici? Come viene calcolata la latenza in lettura di un disco? Come si può mascherare?

12. [5] Le memorie sono costituite da una gerarchia costruita secondo criteri ben precisi. Rispondere a queste domande motivando la risposta:

- Cosa si intende per gerarchia delle memorie?
- Spiegare chiaramente cosa si intenda per **coerenza** e **consistenza** di una memoria. Fare degli esempi. A quali memorie si applicano?
- Quali sono i meccanismi messi in atto per garantire la **coerenza** e la **consistenza** della memoria nelle architetture **mono-processore** e nelle architetture **multi-processore**? Quali sono i vantaggi e svantaggi di ciascun meccanismo?
- Cosa si intende per hit e miss di una memoria cache e come vengono gestiti? Chi li gestisce? Perché le miss sono critiche?
- Che differenza c'è tra una miss e un page fault? Cos'è un page fault?
- Cos'è la memoria virtuale? Cos'è la Tabella delle pagine? Dove si trova? Cos'è il "Translation Lookaside buffer"? Dove si trova? A cosa servono la memoria virtuale, il TLB e la tabella delle pagine?
- Che relazione c'è tra la memoria virtuale e la memoria fisica? Chi utilizza la memoria virtuale? Chi utilizza la memoria fisica? Cosa succede quando la CPU chiede una parola alla memoria?

Registri del register file

| | | | |
|-----|----------------------------|-----|---------------------------|
| 0 | zero constant 0 | 16 | s0 callee saves |
| 1 | at reserved for assembler | ... | (caller can clobber) |
| 2 | v0 expression evaluation & | 23 | s7 |
| 3 | v1 function results | 24 | t8 temporary (cont'd) |
| 4 | a0 arguments | 25 | t9 |
| 5 | a1 | 26 | k0 reserved for OS kernel |
| 6 | a2 | 27 | k1 |
| 7 | a3 | 28 | gp Pointer to global area |
| 8 | t0 temporary: caller saves | 29 | sp Stack pointer |
| ... | (callee can clobber) | 30 | fp frame pointer (s8) |
| 15 | t7 | 31 | ra Return Address (HW) |



