

ARCHITETTURE AVANZATE DEI CALCOLATORI - Prima prova in itinere del 5/11/2005
Compito tipo A

1. Con riferimento alle gerarchie di memoria ed alle cache a mappatura diretta, indicare quali tra le seguenti affermazioni sono corrette.
 - A) Gli accessi a locazioni contigue di memoria sono resi molto efficienti dalla presenza di una cache.
 - B) La lunghezza del campo indice di una cache a mappatura diretta dipende dalla dimensione della cache.
 - C) In una cache a mappatura diretta non si hanno conflict miss.
 - D) Dato un calcolatore con frequenza di clock pari a 1 GHz, caratterizzato da miss rate=0.05, hit time=1 ciclo di clock e miss penalty=20 cicli di clock, il tempo medio di accesso alla memoria è pari a 2 ns.
 - E) Per aumentare la dimensione della cache di un calcolatore è necessario aumentare la dimensione della sua memoria principale.
 - F) In una cache a mappatura diretta con 16K righe e blocchi da 64 bit, installata in un'architettura con indirizzi di memoria a 32 bit, il campo tag ha dimensione pari a 14 bit.

2. Con riferimento alla pipeline a 5 stadi del processore MIPS, indicare quali tra le seguenti affermazioni sono corrette.
 - A) In condizioni ideali (assenza di stalli), il throughput migliora di 5 volte rispetto al processore senza pipeline.
 - B) L'introduzione della pipeline può far aumentare la latenza della singola istruzione.
 - C) Le condizioni per riconoscere le criticità sui dati nel processore MIPS con pipeline considerate a lezione permettono di identificare e risolvere tramite forwarding la seguente criticità:

```
lw $2, 0($5)
sw $2, 0($4)
```
 - D) Nel registro di pipeline MEM/WB non viene scritto nessun segnale di controllo generato dall'unità di controllo.
 - E) Per la gestione di un'eccezione basta asserire tutti i segnali di Flush: così si svuota la pipeline dalle istruzioni caricate dopo quella che ha generato l'istruzione e si preserva il valore dei registri coinvolti nell'istruzione che ha generato l'istruzione.
 - F) Senza anticipazione della decisione sul salto al secondo stadio, l'indirizzo di salto è noto nel terzo stadio, ma il suo valore viene usato nel quarto stadio in caso di branch taken per aggiornare il PC.

3. Con riferimento alle tecniche per aumentare il grado di parallelismo delle istruzioni, indicare quali tra le seguenti affermazioni sono corrette.
 - A) Se in un processore le istruzioni vengono eseguite out-of-order, non possono essere ritirate in-order.
 - B) Nell'algoritmo di Tomasulo le dipendenze di tipo WAR e WAW sono eliminate con la tecnica di ridenominazione dei registri.
 - C) In una esecuzione out-of-order è comunque possibile che sia necessario inserire degli stalli.
 - D) In una esecuzione speculativa vengono avviate istruzioni che non saranno mai ritirate.
 - E) Non è possibile decodificare più istruzioni nello stesso ciclo di clock.
 - F) I processori VLIW hanno un'architettura hardware più complessa rispetto ai processori multiple-issue dinamici.

4. Con riferimento alle criticità sui dati e sul controllo, indicare quali tra le seguenti affermazioni sono corrette.
 - A) Si consideri la pipeline ottimizzata del processore MIPS. Data la sequenza di istruzioni

```
lw $2, 0($5)
add $2, $2, $2
add $2, $2, $2
```

le due criticità sui dati presenti (di tipo load/use e define/use) sono risolvibili con forwarding ed inserendo uno stallo tra la prima e la seconda istruzione.
 - B) Si consideri la pipeline senza ottimizzazioni del processore MIPS. Data la sequenza di istruzioni

```
addi $5, $5, -1
bnez $5, Loop
addi $6, $6, 4
```

essa determina una criticità sui dati tra la prima e la seconda istruzione ed una criticità sul controllo tra la seconda e la terza istruzione (o quella di destinazione del branch), per risolvere le quali è necessario inserire in totale quattro stalli.
 - C) L'indirizzamento della tabella di predizione dei salti avviene prelevando alcuni bit dall'indirizzo dell'istruzione di salto.
 - D) La branch history table ad 1 bit ha un'accuratezza elevata nel caso di cicli brevi.
 - E) Considerando che un branch durante l'esecuzione di un programma ha il comportamento NTTNT ed avendo a disposizione una BHT a 2 bit inizializzata a 10 (weakly predict taken), l'accuratezza della predizione è pari al 40%.
 - F) L'istruzione posta dal compilatore nel delay slot viene eseguita indipendentemente dall'esito del branch.

5. Si consideri il seguente frammento di codice in linguaggio C, dove A e B sono due vettori di interi :

```
for (i=0; i<1000; i++)  
    A[i] = A[i] + B[i];
```

Il sorgente sia stato compilato nel seguente codice in assembler MIPS. Si assuma che i registri \$s0 e \$s1 siano stati inizializzati rispettivamente con l'indirizzo di A[0] e B[0], che \$s2 contenga il valore 1000, e che \$t0 sia inizializzato a 0.

```
Loop: lw $t1, 0($s0)  
      lw $t2, 0($s1)  
      add $t3, $t1, $t2  
      sw $t3, 0($s0)  
      addi $s0, $s0, 4  
      addi $s1, $s1, 4  
      addi $t0, $t0, 1  
      bne $t0, $s2, Loop
```

Si assuma che la criticità tra add e sw possa essere risolta tramite forwarding (spiegare come).

- Schedulare in modo efficiente il ciclo sul processore MIPS con pipeline statica two-issue. Calcolare il valore del CPI ottenuto e confrontarlo con quello ideale.
 - Effettuare il loop unrolling (srotolando il ciclo per 2 volte) e riordinare il codice (eventualmente ridenominando i registri) con l'obiettivo di minimizzare il CPI sulla pipeline ottimizzata del processore MIPS scalare. Calcolare il valore del CPI ottenuto prima e dopo il loop unrolling ed il riordinamento.
6. Sia dato il seguente ciclo di un programma in assembler MIPS. Si supponga che i registri \$4 e \$7 siano stati inizializzati rispettivamente ai valori 0 e 4N. I simboli BASEA, BASEB, BASEC, BASED sono costanti a 16 bit, prefissate.

```
Loop: lw $2, BASEA($4)  
      sw $2, BASEB($4)  
      lw $3, BASEC($4)  
      add $3, $2, $3  
      addi $3, $3, 4  
      sw $3, BASED($4)  
      addi $4, $4, 4  
      bne $4, $7, Loop
```

Si supponga che il ciclo venga iterato N volte (con N grande a piacere) tramite una pipeline MIPS a 5 stadi. Si consideri una generica iterazione del ciclo.

- Si supponga che la pipeline sia priva di ottimizzazioni.
 - Disegnare il diagramma a ciclo multiplo della pipeline ed individuare le criticità sui dati di tipo RAW e le criticità sul controllo.
 - Inserire gli stalli necessari a risolvere le criticità individuate.
 - Calcolare il numero totale di stalli inseriti ed il CPI asintotico ottenuto.
- Si supponga che nella pipeline siano state introdotte le seguenti ottimizzazioni:
 - nel banco dei registri è possibile la lettura e la scrittura nello stesso ciclo di clock;
 - forwarding dei dati (se necessario, è possibile introdurre ulteriori percorsi di forwarding non considerati a lezione, motivandoli opportunamente);
 - anticipazione del salto nello stadio ID.
 - Disegnare il diagramma a ciclo multiplo della pipeline ed individuare le criticità sui dati di tipo RAW e le criticità sul controllo rimaste; indicare i percorsi di forwarding utilizzati.
 - Inserire gli stalli necessari a risolvere le criticità rimaste.
 - Calcolare il numero totale di stalli inseriti ed il CPI asintotico ottenuto.

Esercizio	Punteggio massimo
Teoria (da es. 1 a es. 4)	14
Esercizio 5	9
Esercizio 6	9
Totale	32