Secondo Compitino di Architettura, prof. Chiola

31/5/1996

Cognome :

Nome :

Matricola :

Regole Generali

Questo compito é composto da 2 diversi tipi di domande: alcune a risposte multiple, altre in cui dovete dare voi la risposta corretta mediante lo svolgimento di un semplice esercizio. Nel secondo tipo di domande si richiede di riportare solo il risultato finale.
Le risposte vanno riportate sul MODULO DI RISPOSTA distribuito a parte. Si consiglia comunque di riportare le risposte anche all'interno del testo in modo che il candidato possa tenere una copia dopo la consegna per la verifica delle risposte ed il calcolo del punteggio. Il calcolo dei punteggi relativi ai due tipi di domande é spiegato più avanti.
NON USARE FOGLI ADDIZIONALI né per la consegna del compito né per la "brutta"! E' ammesso utilizzare il rovescio dei fogli del testo come "brutta copia" (che in ogni caso non sarà presa in considerazione per la correzione).
Usare solo penne e NON MATITE per la compilazione delle risposte, in quanto NON VERRANNO ACCETTATE CONTESTAZIONI DI NESSUN TIPO riguardo eventuali correzioni o cancellazioni (comprese quelle effettuate col bianchetto).
Non é ammessa nessuna forma di comunicazione tra i candidati. Non sono ammessi "prestiti" di materiale di nessun tipo tra i candidati.
E' possibile assentarsi dall'aula solo previo deposito del testo del compito presso il docente "di guardia". Anche in caso di ritiro anticipato dal compito il testo dell'esercizio NON PUÒ essere portato fuori dall'aula prima dello scadere del termine di consegna.

Compilazione delle risposte

Si consiglia di scrivere le risposte inizialmente sul testo e di utilizzare gli ultimi 5–10 minuti per ricopiare le risposte sull'apposito foglio da consegnare.

Domande a risposte multiple

Rispondere segnando sul testo con una crocetta le risposte che ritenete corrette e lasciando in bianco quelle che ritenete errate. Ad ogni domanda corrisponde almeno una risposta corretta (ci possono essere più risposte corrette per una domanda).
Domande di questo tipo hanno associati due valori di punteggio: uno negativo e uno positivo. Ogni risposta errata segnata con una crocetta comporta la detrazione del valore negativo dal punteggio complessivo accumulato. Il valore positivo corrisponde al punteggio globalmente assegnato a tutte le risposte corrette ad una domanda.
Per esempio, se una domanda etichettata coi valori $(-1, 4)$ prevede 4 risposte possibili di cui **a. d.** errate e **b. c.** corrette, allora se il candidato segna le risposte **b. c.** e non segna le risposte **a. d.** ottiene il massimo punteggio 4, mentre se segna le risposte **a. b.** e non **c. d.** allora ottiene il punteggio $-1+2=1$; nel caso segnasse tutte le risposte otterrebbe in questo esempio il punteggio $-1+2+2-1=2$. Notare

che in generale, CONTRARIAMENTE a quanto succede nell'esempio, la somma algebrica dei punteggi ottenuti segnando tutte le risposte é ≤ 0!

Esercizi

In questo caso é il candidato che deve calcolare e scrivere la risposta finale. Ogni domanda é accompagnata da un solo valore positivo che indica il punteggio massimo assegnato per la risposta corretta a tutti i quesiti. In caso di risposte parziali o parzialmente corrette il punteggio viene ridotto in proporzionalità diretta rispetto al numero di risposte corrette (con troncamento al più grande numero intero minore o uguale nel caso di risultato frazionario).

Tattica suggerita

Come nel caso di altri test da completarsi in tempi predefiniti, risulta normalmente conveniente completare il compito in due passate: in una prima fase dedicare una quantità predefinita e limitata di tempo alla lettura di tutte le domande, svolgendo solo gli esercizi più semplici. In un secondo passo (dopo aver risposto rapidamente a tutte le domande semplici) tornate a considerare quelle più difficili in ordine decrescente di punteggio. In questo modo riuscirete a dedicare la maggior quantità di tempo alla soluzione degli esercizi difficili ma più redditizi. Per le domande a risposte multiple con punteggi negativi evitate di segnare risposte di cui non siete assolutamente certi in quanto i valori dei punteggi sono studiati per penalizzare i "tentativi a caso". Per gli esercizi senza punteggi negativi invece le risposte sbagliate non vi penalizzeranno.

Calcolo del punteggio

Il voto del secondo compito espresso in quindicesimi sarà calcolato applicando la seguente formula, nella quale s rappresenta la somma dei punti totalizzati per le varie domande:

$$\text{Voto} = 3.45 + \min(s, 27) \times 0.25 + (s - 27) \times 0.25$$

il voto in quindicesimi del primo e del secondo compitino e arrotondando il risultato (a patto che entrambi i compiti siano sufficienti, ovvero che raggiungano almeno il voto di 8/15). Voti complessivi superiori a 30/30 verranno poi considerati equivalenti alla "lode". Voti inferiori a 18/30 verranno considerati insufficienti, e comporteranno la necessità di superare la prova scritta completa in uno dei 5 appelli annuali d'esame. Il voto finale verrà poi stabilito a partire dal voto dello scritto con un riaggiustamento di tale valore fino ad una variazione massima di 4/30 in più o in meno, a seconda dell'esito della prova orale.

Domande

1. (punti: -1, 4)
   Il microsequenziatore della microarchitettura VM-1 vista a lezione (e oggetto del laboratorio):

   (a) é un registro formato da flip flop di tipo J-K che contiene la sequenza delle microistruzioni da eseguire;

   (b) é un circuito sequenziale destinato a mantenere l'indirizzo della cella della Control Store contenente la prossima microistruzione da eseguire;

   (c) é un circuito combinatorio destinato a generare il prossimo valore da inserire nel micro program counter per passare all'esecuzione della microistruzione contenuta nella cella successiva della Control Store;

   (d) é un circuito combinatorio per generare la corretta sequenza delle fasi di clock per comandare i dispositivi sincroni del data path;

   (e) é un circuito sequenziale per la memorizzazione dei valori dei bit N, Z prodotti dall'ALU in modo da consentirne successivamente il test durante l'esecuzione di istruzioni di salto condizionale;

   (f) nessuna delle affermazioni precedenti é corretta.

2. (punti: -1, 4)
   Considerare un insieme di istruzioni di tipo ortogonale di una macchina convenzionale composto da: 6 istruzioni a 3 operandi, 8 istruzioni a 2 operandi, 5 istruzioni a 1 operando e 10 istruzioni senza operandi espliciti.

   (a) Se ciascun operando richiede 2 bit di codifica, le istruzioni possono essere tutte codificate utilizzando sempre 9 bit;

   (b) Se ciascun operando richiede 3 bit di codifica, le istruzioni possono essere tutte codificate utilizzando almeno 11 bit;

   (c) Se ciascun operando richiede 3 bit di codifica, le istruzioni possono essere tutte codificate utilizzando sempre 12 bit;

   (d) Se ciascun operando richiede 4 bit di codifica, le istruzioni possono essere tutte codificate utilizzando almeno 14 bit;

   (e) Se ciascun operando richiede 5 bit di codifica, le istruzioni possono essere tutte codificate utilizzando sempre 17 bit;

   (f) Se ciascun operando richiede 9 bit di codifica, le istruzioni possono essere tutte codificate utilizzando sempre 32 bit.

3. (punti: -1, 4)
   La tecnica di pipeline:

   (a) permette di aumentare la frequenza di completamento delle istruzioni di una macchina convenzionale fino ad un massimo prestabilito e pari al numero di stadi della pipe, a parità di frequenza di clock;

   (b) permette di ridurre il tempo di completamento delle istruzioni di una macchina convenzionale fino ad un massimo prestabilito e pari al numero di stadi della pipe a parità di frequenza di clock;

   (c) impone l'introduzione del concetto di "delay slot" per poter realizzare istruzioni di salto di programma in macchine RISC;

   (d) viene applicata solo in macchine convenzionali di tipo RISC;

(e) non potrebbe essere applicata nella realizzazione della macchina convenzionale VM-2 vista a lezione;

(f) nessuna delle affermazioni precedenti é corretta.

4. (punti: -1, 4)
Il linguaggio VHDL:

(a) é adatto a descrivere sistemi solo a livello L1 (di microarchitettura) ma non a livello L0 (logica circuitale) o L2 (macchina convenzionale);

(b) permette di definire moduli comportamentali (behaviour) generici, la cui interfaccia può variare dinamicamente durante la simulazione del modulo;

(c) permette di definire moduli strutturali generici, la cui interfaccia può variare da istanza ad istanza a seconda del parametro di istanziazione utilizzato;

(d) permette di intercalare moduli definiti solo a livello strutturale a moduli definiti a livello comportamentale, senza generare problemi di simulazione solo a patto che i primi facciano riferimento prima o poi a moduli comportamentali di livello di astrazione più basso;

(e) la realizzazione di dispositivi hardware deve sempre essere specificata in termini di moduli definiti a livello comportamentale;

(f) nessuna delle affermazioni precedenti é corretta.

5. (punti: -1, 4)
Una memoria associativa:

(a) a differenza di una RAM non richiede la conoscenza del contenuto di una cella di memoria per potervi accedere;

(b) a differenza di una RAM non viene normalmente realizzata nella versione "dinamica" (con refresh) ma solo nella versione "statica" per motivi di velocità di accesso;

(c) come una RAM richiede la conoscenza dell'indirizzo di una cella di memoria per potervi accedere;

(d) é normalmente più veloce di una RAM a parità di tecnologia a causa dell'assenza del decoder per l'indirizzamento;

(e) é normalmente più lenta di una RAM a parità di tecnologia a causa della presenza del comparatore per la ricerca associativa;

(f) é normalmente più costosa di una RAM a parità di tecnologia e dimensioni a causa della circuiteria di ricerca associativa e della memorizzazione dei campi "tag."

6. (punti: 6)
La connessione di una memoria cache di primo livello da 32 KByte operante a livello di indirizzi virtuali espressi in byte su 32 bit al variare della organizzazione in linee e del livello di associatività richiede la presenza dei seguenti numeri di bit per il campo "tag":

associatività	dimensione linee	N. bit tag
corr. dir.	4 parole da 4 byte	
2 insiemi	4 parole da 4 byte	
2 insiemi	2 parole da 4 byte	
4 insiemi	2 parole da 4 byte	
8 insiemi	2 parole da 2 byte	
8 insiemi	4 parole da 2 byte	

7.
Considerare un sistema di traduzione da indirizzamento virtuale a indirizzamento fisico realizzato mediante la tecnica di paginazione e segmentazione. Ogni indirizzo logico può far riferimento ad un massimo di 8 segmenti. Ciascun segmento può essere costituito da un massimo di 16 pagine. Ciascuna pagina é costituita da 256 parole di 32 bit. L'indirizzamento é espresso a livello di byte. La memoria fisica può essere costituita al massimo da 32 KByte.

(a) (punti: 2)
Da quanti bit sono costituiti gli indirizzi virtuali e gli indirizzi fisici?

(b) (punti: 2)
Calcolare i valori $v1$ e $v2$ come specificato qui di seguito e rappresentarli in forma binaria sul numero di bit determinato nella risposta (a) per gli indirizzi virtuali:

$$v1 = 1024 * C1 + 16 * C2$$

$$v2 = 512 * N1 + 8 * N2$$

dove $C1$ e $C2$ rappresentano i numeri d'ordine della prima e seconda lettera del cognome mentre $N1$ e $N2$ rappresentano i numeri d'ordine della prima e seconda lettera del nome (Esempio: Giovanni Chiola, $C1 = 3$, $C2 = 8$, $N1 = 7$, $N2 = 9$).

(c) (punti: 8)

Date le seguenti tabelle di traduzione:

tab. segmenti

n. segm	n. pag.	tab. pag.
0	7	tab0
1	5	tab1
2	1	tab2
3	0	—
4	0	—
5	0	—
6	0	—
7	0	—

tab0

n.pag.log.	n.pag.fis
0	4
1	6
2	8
3	10
4	9
5	11
6	12

tab1

n.pag.log.	n.pag.fis
0	2
1	1
2	3
3	5
4	7

tab2

n.pag.log.	n.pag.fis
0	0

Interpretando $v1$ e $v2$ determinati nella domanda (b) come indirizzi virtuali, tradurli nei corrispondenti indirizzi fisici $f1$ e $f2$.

8. (punti: 14)
Considerare le seguenti due procedure scritte in Assembler SPARC:

```
proc1:  save %sp,-104,%sp
        mov 1,%l0
L4:     ldub [%i0],%o0
        cmp %o0,0
        be L3
```

```
        andcc %l0,%i1,%g0
        be L5
        add %i0,1,%i0
        call _print_char
        nop
L5:     b L4
        sll %l0,1,%l0
L3:     call _print_new_line
        nop
        ret
MANCA1:

proc2:  save %sp,-104,%sp
        mov 1,%l0
        mov 1,%l1
        mov %i0,%l2
L10:    ldub [%l2],%o0
        cmp %o0,0
        be L18
        add %l2,1,%l2
        b L10
        sll %l1,1,%l1
L15:    call proc1
MANCA2:
        add %l0,1,%l0
L18:    cmp %l0,%l1
        blu L15
        mov %i0,%o0
        ret
        restore
```

Completare con le due istruzioni mancanti identificate dalle etichette `MANCA1` e `MANCA2`.

9. (punti: 18)
   Sia dato il seguente programma in codice macchina VM-2, traduzione della procedura `proc2` definita in assembler SPARC nella domanda precedente:

```
proc2:   LOC8 1
         PUSH
         PUSH
         LODL 1
         PUSH
         PSHI
su:      POP
         JZER qui
         LODL -3
I1:
         STOL -3
         LOC8 1
         ADDL -4
```

```
I2:
        JUMP su
qui:    LODL -3
        SIGN
        ADDL -2
        JNEG giu
        RETN
giu:    LODL 1
        PUSH
        LODL -2
        PUSH
        CALL proc2
I3:
        LOC8 1
        ADDL -2
        STOL -2
        JUMP qui
```

Completare con le 3 istruzioni mancanti (individuate dalle etichette `I1`, `I2` e `I3`).

Risposte Compitino di Architettura, 31/5/1996

Cognome : **Nome:** **Matricola:**

1. (punti: -1, 4) **a** **b** **c** **d** **e** **X f**
2. (punti: -1, 4) **a** **b** **X c** **d** **e** **X f**
3. (punti: -1, 4) **X a** **b** **c** **d** **e** **f**
4. (punti: -1, 4) **a** **b** **X c** **X d** **e** **f**
5. (punti: -1, 4) **a** **X b** **c** **d** **e** **X f**
6. (punti: 3)

associatività	dimensione linee	N. bit tag
corr. dir.	4 parole da 4 byte	17
2 insiemi	4 parole da 4 byte	18
2 insiemi	2 parole da 4 byte	18
4 insiemi	2 parole da 4 byte	19
8 insiemi	2 parole da 2 byte	20
8 insiemi	4 parole da 2 byte	20

7.
   (a) (punti: 2) virtuali 17, fisici 15

   (b) (punti: 2) Esempio: Giovanni Chiola
   v1= 000 0011 0010000000, v2= 000 0011 1001001000

   (c) (punti: 8) f1= 1010 0010000000, f2= 1010 1001001000

8. (punti: 14)

```
        ...
        ret
MANCA1: restore     ******************************

        ...
L15:    call proc1
MANCA2: mov %l0,%o1 ******************************
        add %l0,1,%l0
        ...
```

9. (punti: 18)

```
 ...
        LODL -3
I1:     ADDL -3   oppure LSHF  ********************
        STOL -3
        LOC8 1
        ADDL -4
I2:     STOL -4                ********************
        JUMP su
 ...
        CALL proc2
I3:     INSP 2                 ********************
        LOC8 1
 ...
```

Risposte Compitino di Architettura, 31/5/1996

Cognome : **Nome:** **Matricola:**

1. (punti: -1, 4) **a** **b** **c** **d** **e** **f**
2. (punti: -1, 4) **a** **b** **c** **d** **e** **f**
3. (punti: -1, 4) **a** **b** **c** **d** **e** **f**
4. (punti: -1, 4) **a** **b** **c** **d** **e** **f**
5. (punti: -1, 4) **a** **b** **c** **d** **e** **f**
6. (punti: 6)

associatività	dimensione linee	N. bit tag
corr. dir.	4 parole da 4 byte	
2 insiemi	4 parole da 4 byte	
2 insiemi	2 parole da 4 byte	
4 insiemi	2 parole da 4 byte	
8 insiemi	2 parole da 2 byte	
8 insiemi	4 parole da 2 byte	

7.
   (a) (punti: 2) bit virtuali= bit fisici=
   (b) (punti: 2) v1= v2=
   (c) (punti: 8) f1= f2=

8. (punti: 14)

```
        ret
MANCA1:


 ...
L15:    call proc1
MANCA2:
        add %l0,1,%l0
```

9. (punti: 18)

```
         LODL -3
I1:
         STOL -3
         LOC8 1
         ADDL -4
I2:
         JUMP su
 ...
         CALL proc2
I3:
         LOC8 1
```