

Prestazioni CPU

Corso di Calcolatori Elettronici A

2007/2008

Sito Web: <http://prometeo.ing.unibs.it/quarella>

Prof. G. Quarella

prof@quarella.net

Prestazioni

- Si valutano in maniera diversa a seconda dell'applicazione d'interesse
- Le grandezze da considerare sono:
 - **Tempo di risposta o di esecuzione**
 - **Throughput**

Es.: Un processore più veloce migliora il tempo di risposta e il throughput.

Aggiungere un processore uguale incrementa il throughput ed eventualmente il tempo di risposta se i lavori da eseguire saturavano il throughput fino a creare code.

Prestazioni

Computer ↔ Applicazione

- Desktop computer
 - Tempo di risposta
- Server
 - Throughput e tempo di risposta
- Embedded computer
 - Vincoli real-time sul tempo di risposta

Costo e consumo energetico sono altri due fattori molto importanti nel progetto di un calcolatore.

Prestazioni di un'applicazione

- L'algoritmo
- Il linguaggio di programmazione o il compilatore
- Il sistema operativo
- **Il processore**
- Il sistema I/O e le periferiche

Prestazioni e tempo di esecuzione

- Per una macchina X vale la relazione

$$\text{prestazioni}_X = \frac{1}{\text{tempo di esecuzione}_X}$$

- Se per due macchine X e Y, le prestazioni di X sono migliori di Y si ha che

$$\frac{\text{prestazioni}_X}{\text{tempo di esecuzione}_X} > \frac{\text{prestazioni}_Y}{\text{tempo di esecuzione}_Y}$$

$$\text{tempo di esecuzione}_Y > \text{tempo di esecuzione}_X$$

Relazione fra le prestazioni di 2 macchine

- Se la macchina X è n volte più veloce della macchina Y si scriverà:

$$\frac{\text{prestazioni}_X}{\text{prestazioni}_Y} = n$$

- E quindi...

$$\frac{\text{prestazioni}_X}{\text{prestazioni}_Y} = \frac{\text{tempo di esecuzione}_Y}{\text{tempo di esecuzione}_X} = n$$

Tempo di CPU

Per quantificare le prestazioni di un calcolatore utilizzeremo il tempo di esecuzione della CPU o tempo di CPU ovvero il tempo speso dalla CPU nell'eseguire un determinato programma.

Il tempo speso in operazioni di I/O o per eseguire altri programmi non è considerato.

Il tempo di risposta percepito dall'utente sarà il tempo trascorso nell'esecuzione del programma e non il tempo di CPU.

Relazioni tra le metriche

- Ci occuperemo delle prestazioni della CPU e quindi del tempo di esecuzione della CPU
- Valgono le seguenti relazioni:

$$\begin{array}{l} \text{Tempo di CPU} \\ \text{relativo a un programma} \end{array} = \begin{array}{l} \text{cicli di clock della CPU} \\ \text{relativi al programma} \end{array} \times \begin{array}{l} \text{periodo di ciclo} \\ \text{del clock} \end{array}$$

$$\begin{array}{l} \text{Tempo di CPU} \\ \text{relativo a un programma} \end{array} = \frac{\text{cicli di clock della CPU relativi al programma}}{\text{frequenza di clock}}$$

Cicli di clock della CPU e Cicli di clock per istruzione

- **CPI (clock per istruzione)** indica il **numero medio di cicli di clock per istruzione**, calcolato come la media del numero di cicli di clock che le diverse istruzioni richiedono per essere completate in un programma.
- Valgono quindi le seguenti relazioni:

$$\begin{array}{l} \text{cicli di clock} \\ \text{della CPU} \end{array} = \begin{array}{l} \text{numero di istruzioni} \\ \text{nel programma} \end{array} \times \text{CPI}$$

Cicli di clock della CPU e Cicli di clock per istruzione

tempo di CPU = numero di istruzioni × CPI × durata del ciclo di clock

o alternativamente...

$$\text{tempo di CPU} = \frac{\text{numero di istruzioni} \times \text{CPI}}{\text{frequenza di clock}}$$

Esempio

- Siano date due implementazioni diverse dello stesso set di istruzioni
- Il calcolatore A ha una durata di ciclo di clock pari a 1 ns e un CPI pari a 2 per un determinato programma
- Il calcolatore B ha una durata del ciclo di clock pari a 2 ns e un CPI pari a 1,2 per lo stesso programma
- Quale dei due è il più veloce? (sia I il numero di istruzioni del programma)

$$\text{cicli di clock della CPU}_A = I \times 2$$

$$\text{cicli di clock della CPU}_B = I \times 1,2$$

$$\text{tempo di CPU}_A = \text{cicli di clock della CPU}_A \times \text{periodo del ciclo di clock}_A = I \times 2 \times 1 \text{ ns} = 2 I \text{ ns}$$

$$\text{tempo di CPU}_B = I \times 1,2 \times 2 \text{ ns} = 2,4 I \text{ ns}$$

$$\frac{\text{Prestazioni della CPU}_A}{\text{Prestazioni della CPU}_B} = \frac{\text{Tempo di esecuzione}_B}{\text{Tempo di esecuzione}_A} = \frac{2,4 I \text{ ns}}{2 I \text{ ns}} = \mathbf{1,2}$$

la CPU di A è 1,2 volte più veloce

Esercizio

Si supponga di disporre di due macchine: M_1 con frequenza di clock pari a 200 MHz e M_2 con frequenza di clock pari a 300 MHz; e si consideri un programma tale che:

	su M_1	su M_2
Tempo	10 sec	5 sec
Istruzioni eseguite	200×10^6	160×10^6

Trovare i CPI per eseguire il programma su ciascuna macchina

$$T_{M1} = I_{M1} \times \text{CPI}_{M1} / f_{M1} = 10 \text{ sec} = \frac{200 \cdot 10^6 \cdot \text{CPI}_{M1}}{200 \cdot 10^6 \text{ cicli/sec}} \quad \text{CPI}_{M1} = 10 \text{ cicli}$$

$$T_{M2} = I_{M2} \times \text{CPI}_{M2} / f_{M2} = 5 \text{ sec} = \frac{160 \cdot 10^6 \cdot \text{CPI}_{M2}}{300 \cdot 10^6 \text{ cicli/sec}} \quad \text{CPI}_{M2} = 9,375 \text{ cicli}$$

(durata ciclo clock $_{M1} = 1 / (200 \times 10^6 \text{ cicli/sec}) = 5 \text{ ns}$
durata ciclo clock $_{M2} = 1 / (300 \times 10^6 \text{ cicli/sec}) = 3,3 \text{ ns}$)

Un'altra equazione per i cicli di clock della CPU

È possibile calcolare il numero dei cicli di clock della CPU anche analizzando i diversi tipi di istruzioni e utilizzando per ciascuno il relativo numero di cicli di clock.

Si usa in questo caso:

$$\text{cicli di clock della CPU} = \sum_{i=1}^n (\text{CPI}_i C_i)$$

Dove C_i è il numero di istruzioni della classe i che sono state eseguite e CPI_i è il numero medio di cicli per le istruzioni della classe i , mentre n è il numero delle classi

Esercizio

I progettisti hardware di un certo calcolatore hanno fornito le seguenti informazioni

Classe di istruzioni	CPI
A	1
B	2
C	3

Un progettista di compilatori deve scegliere tra due sequenze di codici

Sequenza di codici	Numero di istruzioni per ciascuna classe		
	A	B	C
1	2	1	2
2	4	1	1

Quale sequenza di codici causa l'esecuzione del maggior numero di istruzioni?

Quale viene eseguita più velocemente? Qual è il CPI per ciascuna delle sequenze?

Soluzione

- Sequenza 1: $2 + 1 + 2 = 5$ istruzioni
Sequenza 2: $4 + 1 + 1 = 6$ istruzioni
→ Il numero minore di istruzioni si ottiene con la sequenza 1

- Usando la formula: cicli di clock della CPU = $\sum_{i=1}^n (\text{CPI}_i C_i)$

Cicli di clock della CPU₁ = $(2 \cdot 1) + (1 \cdot 2) + (2 \cdot 3) = 2 + 2 + 6 = 10$ cicli

Cicli di clock della CPU₂ = $(4 \cdot 1) + (1 \cdot 2) + (1 \cdot 3) = 4 + 2 + 3 = 9$ cicli

→ La sequenza 2 è più veloce

- Dato che: $\text{CPI} = \text{cicli di clock della CPU} / \text{numero di istruzioni}$

$\text{CPI}_1 = \text{cicli di clock della CPU}_1 / \text{numero di istruzioni}_1 = 10/5 = 2$

$\text{CPI}_2 = \text{cicli di clock della CPU}_2 / \text{numero di istruzioni}_2 = 9/6 = 1,5$

Esercizio

- Si considerino due diverse implementazioni, M1 e M2, dello stesso set di istruzioni, cui appartengono 4 diverse classi di istruzioni (A, B, C, D)
- M1 ha frequenza di clock pari a 500 MHz e il numero medio di cicli di clock per ciascuna classe di istruzioni è dato dalla tabella A
- M2 ha frequenza di clock pari a 750 MHz e il numero medio di cicli di clock per ciascuna classe di istruzioni è dato dalla tabella B

Classe di istruzioni	CPI
A	1
B	2
C	3
D	4

Tabella A

Classe di istruzioni	CPI
A	2
B	2
C	4
D	4

Tabella B

Se il numero di istruzioni di un dato programma è suddiviso in parti uguali fra le classi di istruzioni, di quanto M2 è più veloce di M1?

Soluzione

- Se le istruzioni del programma sono suddivise in parti uguali rispetto alle 4 classi significa che per ogni classe ci sono il 25% delle istruzioni

- Si ricordi inoltre che:

prestazioni = 1 / tempo di esecuzione

tempo di esecuzione = (numero istruzioni × CPI) / frequenza di clock

$$\frac{T_{M1}}{T_{M2}} = \frac{\frac{I \cdot (25\% \cdot 1 + 25\% \cdot 2 + 25\% \cdot 3 + 25\% \cdot 4)}{500 \cdot 10^6 \text{ cicli/sec}}}{\frac{I \cdot (25\% \cdot 2 + 25\% \cdot 2 + 25\% \cdot 4 + 25\% \cdot 4)}{750 \cdot 10^6 \text{ cicli/sec}}} = 1,5 \cdot \frac{0,25 + 0,5 + 0,75 + 1}{0,5 + 0,5 + 1 + 1} = 1,25$$

Quindi M2 è 1,25 volte più veloce di M1

MIPS

- I **MIPS** (**milioni di istruzioni al secondo**) costituiscono un'alternativa al tempo per valutare le prestazioni

$$\text{MIPS} = \frac{\text{numero di istruzioni}}{\text{tempo di esecuzione} \times 10^6}$$

- I MIPS rappresentano una **misura delle istruzioni eseguite**
- Quindi esprimono il concetto di prestazione in modo inverso al tempo di esecuzione: le macchine più veloci hanno un valore più elevato in termini di MIPS
- Il vantaggio dell'uso dei MIPS è dato dal fatto che rappresentano un concetto facile da comprendere: è intuitivo pensare che le macchine più veloci abbiano un valore di MIPS più elevato

Esempio di uso dei MIPS

- 2 Implementazioni di una macchina:
 - Una dotata di una unità per il trattamento dei numeri in virgola mobile (MVM)
 - Una che non possiede questa unità (MSVM)
- Si consideri un programma con la seguente combinazione di operazioni

moltiplicazioni in virgola mobile	10%
somme in virgola mobile	15%
divisioni in virgola mobile	5%
operazioni su interi	70%
- La macchina MVM richiede il seguente numero di cicli di clock per ciascuna classe di istruzioni

moltiplicazioni in virgola mobile	6
somme in virgola mobile	4
divisioni in virgola mobile	20
operazioni su interi	2
- La macchina MSVM emula le operazioni in virgola mobile con operazioni fra interi che richiedono 2 cicli di clock. Il numero delle istruzioni per emulare le operazioni in virgola mobile è:

moltiplicazioni in virgola mobile	30
somme in virgola mobile	20
divisioni in virgola mobile	50

Esempio di uso dei MIPS

Entrambe le macchine hanno frequenza di clock pari a 1000 MHz.
Trovare il valore dei MIPS delle due macchine.

$$\text{MIPS} = \frac{\text{numero di istruzioni}}{\text{tempo di esecuzione} \cdot 10^6}$$
$$\text{tempo di esecuzione} = \frac{\text{cicli di clock della CPU}}{\text{frequenza di clock}}$$

$$\text{MIPS}_{\text{MVM}} = \frac{I}{\frac{I \cdot (0,10 \cdot 6 + 0,15 \cdot 4 + 0,05 \cdot 20 + 0,7 \cdot 2)}{1000 \cdot 10^6} \cdot 10^6} \approx 278$$

$$\text{MIPS}_{\text{MSVM}} = \frac{I}{\frac{I \cdot (0,10 \cdot 60 + 0,15 \cdot 40 + 0,05 \cdot 100 + 0,7 \cdot 2)}{1000 \cdot 10^6} \cdot 10^6} \approx 54$$

Problemi nell'utilizzo dei MIPS

1. Questa misura non tiene conto delle caratteristiche delle istruzioni e quindi non si possono confrontare macchine con set di istruzioni diversi
2. Nello stesso calcolatore i MIPS variano a seconda del programma
3. I MIPS possono fallire nel dare una reale misura delle prestazioni

In conclusione è sempre preferibile usare il tempo di esecuzione come misura di prestazioni

Esempio: confronto fra MIPS e tempo di esecuzione come misura delle prestazioni

- Si consideri una macchina con 3 classi di istruzioni e le seguenti misure di CPI

Classe di istruzioni	CPI
A	1
B	2
C	3

- Si supponga di misurare il codice dello stesso programma generato da due diversi compilatori e di ottenere i seguenti dati

Codice da	Numero di istruzioni (in miliardi) per ciascuna classe di istruzioni		
	A	B	C
Compilatore 1	5	1	1
Compilatore 2	10	1	1

Esempio: confronto fra MIPS e tempo di esecuzione come misura delle prestazioni

- Sia 500 MHz la frequenza di clock della macchina
- Quale sequenza di codice sarà eseguita più velocemente?

Misura con il tempo di esecuzione:

$$\text{cicli di clock della CPU}_1 = (5 \cdot 1 + 1 \cdot 2 + 1 \cdot 3) \cdot 10^9 = 10 \cdot 10^9$$

$$\text{cicli di clock della CPU}_2 = (10 \cdot 1 + 1 \cdot 2 + 1 \cdot 3) \cdot 10^9 = 15 \cdot 10^9$$

$$\text{tempo di esecuzione}_1 = \frac{\text{cicli di clock della CPU}}{\text{frequenza di clock}} = \frac{10 \cdot 10^9}{500 \cdot 10^6} = 20 \text{sec}$$

$$\text{tempo di esecuzione}_2 = \frac{15 \cdot 10^9}{500 \cdot 10^6} = 30 \text{sec}$$

Esempio: confronto fra MIPS e tempo di esecuzione come misura delle prestazioni

Misura con i MIPS

$$\text{MIPS} = \frac{\text{numero di istruzioni}}{\text{tempo di esecuzione} \cdot 10^6}$$

$$\text{MIPS}_1 = \frac{(5+1+1) \cdot 10^9}{20 \cdot 10^6} = 350$$

$$\text{MIPS}_2 = \frac{(10+1+1) \cdot 10^9}{30 \cdot 10^6} = 400$$

Il codice generato dal compilatore 2 ha un valore di MIPS più elevato anche se viene eseguito più lentamente (30 sec contro 20 sec)!

Legge di Amdahl

- Detta anche *legge dei rendimenti decrescenti*
- Se si migliora un aspetto/componente di una macchina, il possibile incremento delle prestazioni è **limitato** dall'ammontare dell'utilizzo dell'aspetto in questione, cioè:

Tempo di esecuzione dopo il miglioramento =

$$\frac{\text{tempo di esecuzione influenzato dal miglioramento}}{\text{ammontare del miglioramento}} + \text{tempo di esecuzione non influenzato}$$

- Questo significa che l'aumento delle prestazioni non è in genere proporzionale al miglioramento di una delle componenti della macchina

La legge di Amdahl ricorda che l'opportunità di operare un miglioramento dipende dal tempo complessivo occupato dall'evento: rendere veloce l'evento più frequente contribuisce a migliorare le prestazioni molto di più che ottimizzare gli eventi rari.

Esempio

- Si supponga di aver migliorato una macchina in modo che le operazioni in virgola mobile vengano svolte 5 volte più velocemente
- Se prima del miglioramento il tempo di esecuzione di un dato benchmark era di 10 secondi quale sarà il tempo di esecuzione dopo il miglioramento supponendo che l'esecuzione delle operazioni in virgola mobile occupasse la metà dei 10 secondi?

$$\text{Tempo dopo il miglioramento} = \frac{5 \text{ secondi}}{5} + 5 \text{ secondi} = 6 \text{ secondi}$$

$$\text{Speedup} = \frac{\text{prestazioni dopo il miglioramento}}{\text{prestazioni prima del miglioramento}} = \frac{\text{tempo prima del miglioramento}}{\text{tempo dopo il miglioramento}} = \frac{10}{6} = 1,67$$

Le prestazioni migliorano solo di un fattore 1,67 e non di un fattore 5.

I benchmark

- Insieme di programmi scelti per misurare le prestazioni
- Insiemi differenti in base alle differenti classi di applicazioni
- Misurazioni registrate in un report così come tutti le informazioni necessarie per garantire la riproducibilità del risultato.
- SPEC (System Performance Evaluation Corporation)

Riferimenti

Computer Organization and Design

The Hardware/Software Interface 3rd Edition

David A. Patterson, John L. Hennessy

Capitolo 4

Versione italiana:

Struttura e Progetto dei Calcolatori

L'Interfaccia Hardware-Software

2^a edizione Zanichelli

<http://en.wikipedia.org/> o <http://it.wikipedia.org/>