

Valutazione delle Prestazioni

Riassunto

Prestazione

- Prestazione = $1 / \text{Tempo di esecuzione}$
- $P_x/P_y = (1+n/100)$
 - n è al percentuale di incremento di P_x risp. a P_y
- In generale: $A/B = \text{ratio}$
 - ratio misura il miglioramento/peggioramento di A rispetto a B
 - Ratio = 1 -> nessun miglioramento
 - Ratio > 1 -> A è più grande di B
 - Ratio < 1 -> A è più piccolo di B

CPI, Clock & MIPS

- $FClock = 1/TClock$
- $CPI = \#clock \text{ per istruzione}$
 - $CPI_{medio} = \#TotClock / \#TotIstr$
- $TIstr = TClock * CPI$
- $TotTime = CPI * \#TotIstr * TClock$
- $MIPS = FClock / (CPI * 10^6)$
 - $= 1/(TClock * CPI * 10^6) = 1/(TIstr * 10^6)$

Media Pesata

- Dati un insieme di valori $\{x_1, \dots, x_n\}$ e le relative frequenze $\{f_{x_1}, \dots, f_{x_n}\}$
 - Somma $\sum f_{x_i} = 1$
 - Media = somma di $(x_i * f_{x_i})$
- Ex:
 - $X = \{ 2, 1, 6, 4 \}$
 - $F_x = \{ 0.2, 0.4, 0.1, 0.3 \}$
 - $M = 2*0.2 + 1*0.4 + 6*0.1 + 4*0.3 = 2.6$
- $f_{x_i} = \#casiX_i / \#TotCasi$
 - Ex: $F_{ALUOp} = \#IstrAlu / \#TotIstr$

Speed-Up

- **Speed up = P_{new} / P_{old}**
= $(1/T_{new}) / (1/T_{old}) = T_{old} / T_{new}$

Nota: con le dovute cautele, si possono usare anche i tempi medi

- **Ex:**
 - $P = (40\% \text{ ALU}, 50\% \text{ LW/SW}, 10\% \text{ JUMP})$
 - $CPICPU1 = \{ALU=2, LW/SW=7, JUMP=1\}$
 - $CPICPU2 = \{ALU=4, LW/SW=6, JUMP=2\}$
 - $CPICPU1 = 2*0.4 + 7*0.5 + 1*0.1 = 4.4 \text{ i/cl}$
 - $CPICPU2 = 4*0.4 + 6*0.5 + 2*0.1 = 3.8 \text{ i/cl}$
 - **SpeedUp = $CPICPU1/CPICPU2 = 4.4 / 3.8 = 1.158$**

Legge di Amdhal

Il miglioramento delle prestazioni globali ottenuto con un miglioramento particolare (e.g. un'istruzione), dipende dalla frazione di tempo in cui il miglioramento era eseguito.

Corollario della legge di Amdhal

Se un miglioramento è utilizzabile solo per una frazione del tempo di esecuzione complessivo (F_m), allora non è possibile accelerare l'esecuzione più del reciproco di uno meno tale frazione:

$$\mathbf{SpeedUpMax} < 1/(1-F_m).$$

- $P = (40\% \text{ ALU}, 50\% \text{ LW/SW}, 10\% \text{ JMP})$
- $CPICPU1 = \{ \text{ALU} = 2 \text{ cl/i}, \text{LW/SW} = 7 \text{ cl/i}, \text{JMP} = 1 \text{ cl/i} \}$
- $FM = (CPI_{ALU} * F_{ALU}) / (CPICPU1) = 0.8/4.4 = 0.18$
- **$SpeedUpMax < 1 / (1 - 0.18) = 1.22$**
- $CPICPU2 = \{ \text{ALU} \approx 0 \text{ cl/i}, \text{LW/SW} = 7 \text{ cl/i}, \text{JMP} = 1 \text{ cl/i} \}$
- $CPICPU2 \approx 0 * 0.4 + 7 * 0.5 + 1 * 0.1 = 3.6 \text{ i/cl}$
- **$SpeedUp = CPICPU1 / CPICPU2 \approx 4.4 / 3.6 = 1.22$**

Nota: Il teorema parla **frazione di tempo!!!** non di percentuale di operazioni:
nell'esempio precedente la frazione di tempo è la frazione di tempo che la CPU dedica alle operazioni ALU cioè il tempo medio richiesto da una OpALU * la percentuale di OpAlu fratto il tempo medio di una generica Op

SpeedUpGlobale

Data F_m , la frazione di tempo di codice ottimizzabile e S_p , lo speedup ottenibile su quel codice:

$$\mathbf{SpeedUpGlobale = 1/(1-F_m+F_m/S_p).}$$

Cache: i tempi

- **HIT_TIME**: Tempo di accesso alla cache.
- **MISS_PENALTY**: è composto da:
 - TEMPO DI TRASFERIMENTO: per trasferire le altre parole del blocco al livello superiore.
 - TEMPO DI ACCESSO: per accedere alla prima parola del blocco dopo che è stato rilevato il fallimento.
- **MISS_TIME** = HIT_TIME + MISS_PENALTY
- **HIT_RATE** = #HIT/#Accessi
- **MISS_RATE** = #MISS/#Accessi
- **Tmedio** = HIT_TIME + MISS_RATE * MISS_PENALTY
- **TCPU** = (#Cicli della CPU in esecuzione + #Cicli di stallo) * TClock
- **CPI_{con_cache}** = $CPI_{exec} + (\#Accessi / \#Istr) * MISS_RATE * MISS_PENALTY$