Lec-3

(*) 2nd class এর Topic Hold করে - আমকে Performance এর chapter পড়ব।

(*) Define Performance.

= Multiple মিশিন এর মধ্যে compare এর জন্য কিছু Criteria দিয়ে তুলনা করা হয়।

অর্থাৎ Criteria অনুযায়ী যে best তার permonce best.

(*) So Computer performance এর জন্য 2টি Criteria :-

① Response Time: কোন একটা কাজ করতেই আমার computer Response করতে কি পরিমান সময় নেয়। অর্থাৎ কোনো কাজ complete করতে যে সময় নেয় সেটিই হলো Response time. [ex: execution time etc]

(এর Response time Late সে তত খারাপ

② Throughput : Persecond (per Unit) এ আমরা computer যে পরিমাণ কাজ করতে পারে সেটিই হচ্ছে throughput.

(যে per second এ বেশি কাজ করে সে ভালো।) [Clock Rate, CPU Time etc..]

* Response time কম হলে ভালো।
* Throughput বেশি হলে ভালো।

[Response time and throughput দিয়ে আমরা 2টি computer এর performance আলাদা করতে পারি।]

Relative Performance:

$$P \propto \frac{1}{R.T.}$$

P = Performance

R.T = Response time or Execution Time

$$P \propto T$$

T = Throughput

RT = ET = CPU Time

- RT → Response Time
- ET → Execution Time

T = CF = CR → Clock Rate

- CF → Clock Frequency

Performance of X

$$\text{Performance of } X = \frac{P_x}{P_y} = \frac{E_y}{E_x}$$

$E_y > E_x$ → Always True

$\Leftrightarrow$ (Response time / Execution Time)

Performance of Y

$$\frac{P_x}{P_y} = \frac{CR_x}{CR_y} \Leftrightarrow (\text{Throughput} / \text{Clock Rate})$$

* Question: Time taken to run a Program :-
  - 10 s on A computer, 15 s on B

so which one has better performance or faster?

Ans:

$$\frac{P_a}{P_b} = \frac{E_b}{E_a}$$

এই পার্থক্য Always এর সময় বেশি একে ছোট হয়।

এই পার্থক্য Right side এ বসে। অর্থাৎ,

> যে faster তাকে slower দ্বারা ভাগ $= \frac{P_A}{P_B} = \frac{E_{T_B}}{E_{T_A}}$ ***

So,

$$\frac{P_a}{P_b} = \frac{E_b}{E_a} = \frac{15}{10} = 1.5$$

$$\Rightarrow P_a = 1.5 P_b$$

(*) So, a is 1.5 time faster than b.

Measuring Execution Time

✳ Elasped Time: একটা কাজ সম্পন্ন করতে computer এর time সহ এর বাইরের সকল কাজ এর time মিলিয়ে total response time যেটা হলো Elasped-time.

[Total response time, including all aspects.]

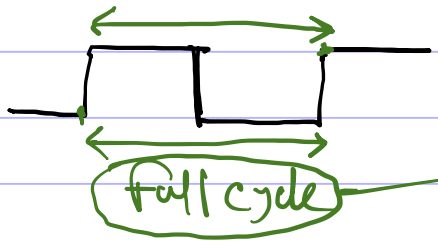
✳ CPU Time: Same ভাবেই একটা কাজ করতে শুধুমাত্র computer এর যে সময় লাগে সেটাই cpu time

[Elasped time → CPU time included.]

CPU clocking

(*) **Clock Period:** একটা clock cycle সম্পন্ন করতে যতটুকু সময় লাগে তাই হচ্ছে clock period.

→ **Clock cycle:** নির্দিষ্ট point থেকে শুরু করে ঐ point এ ফিরে আসতে একটা cycle.



→ full cycle এ যতক্ষন সময় লাগে তাই হচ্ছে clock period.

(*) **Clock Frequency (Rate):** 1 sec এ যতগুলো clock cycle execute হয় তাই হচ্ছে clock Rate.

$\therefore CP = \frac{1}{CR}$

C.R = clock Rate or clock frequency (CF)

C.P = clock Period

CPU Time

(*) কতগুলো instruction কে সম্পন্ন করতে যে পরিমাণে সময় লাগে তাই হলো CPU Time.

So,

(*) $$\text{CPU Time} = \frac{\text{CPU clock cycles}}{} \times \text{Clock cycle Time}$$

We know that,

$$\text{Clock Period} = \text{Clock cycle time} = \frac{1}{\text{clock Rate}}$$

So,

$$\text{CPU Time} = \frac{\text{CPU clock cycles}}{} \times \frac{1}{\text{clock Rate}}$$

(*) $$\Rightarrow \text{CPU Time} = \frac{\text{CPU clock cycles}}{\text{CPU clock Rate}}$$

* Example:

→ Computer A: 2 GHz clock, 10s CPU Time

→ Design computer B:

- Aim for 6s CPU Time
- Can do faster clock, but causes 1.2x clock cycles

→ How fast must Computer B clock be?

Solution:

For Computer A:

$$\text{Clock Rate}_A = 2\ \text{GHz}$$

$$\text{CPU Time}_A = 10\ \text{s}$$

we know that,

$$\text{CPU Time}_A = \frac{\text{clock cycle}_A}{\text{clock Rate}_A}$$

$\Rightarrow$ Clock Cycle$_A$ = CPU Time$_A$ $\times$ Clock Rate$_A$

$= 10 \times 2 = 20$

For Computer B:

CPU Time$_B$ = 6s

Clock Cycle$_B$ = 1.2×20

=

$$\text{CPU Time}_B = \frac{\text{Clock cycle}_B}{\text{Clock Rate}_B}$$

$$\Rightarrow \text{clock Rate}_B = \frac{\text{Clock cycle}_B}{\text{CPU Time}_B}$$

$$= \frac{1.2 \times 10}{6}$$

$= 9$ GHz

$$\frac{\text{faster}}{\text{slower}} = \frac{P_B}{P_A} = \frac{CR_B}{CR_A} = \frac{4}{2} = 2$$

$$\Rightarrow P_B = 2P_A$$

→ Computer B is 2 time faster than computer A.

Instruction Count and CPI

* Clock Cycles = Instruction count (I.C.) × Cycles per Instruction

* CPU Time = Instruction count (I.C.) × C.P.I. × Clock Cycle Time

= $\frac{1}{\text{Clock Rate}}$

CPI → Clock Per Instruction

$$\Rightarrow \text{CPU Time} = \frac{\text{Instruction Count (I.C)}}{} \times \text{C.P.I.} \times \frac{1}{\text{Clock Rate}}$$

$$\text{CPU Time} = \frac{\text{I.C.} \times \text{CPI}}{\text{clock rate}}$$

Example:

- Computer A: Cycle Time = 250ps; CPI = 2.0
- Computer B: Cycle Time = 500 ps; CPI = 1.2
- Both computer has same ISA
- => So, which computer is faster and how much ?

Solution:

We know that,

CPU Time = IC × CPI × Cycle Time

Here, $IC_A = IC_B = I$

So, For A:

$$CPU\ Time_A = IC_A \times CPI_A \times Cycle\ Time_A$$

$$= I \times 250 \times 2$$

$$\therefore CPU\ Time_A = 500\ I$$

For B:

$$CPU\ Time_B = IC_B \times CPI_B \times Cycle\ Time_B$$

$$= I \times 500 \times 1.2$$

$$= 600\ I$$

So,

$$\text{Faster / Slower} = \frac{P_B}{P_A} = \frac{CPU\ Time_B}{CPU\ Time_A} = \frac{600\ I}{500\ I} = 1.2$$

$\Rightarrow$ CPU Time$_B$ = 1.2 × CPU Time$_A$

So, computer B is 1.2 time faster than computer A. (Ans)

More About CPI

(*) If different instruction classes takes different number of cycles:-

[যদি ভিন্ন instruction classes ভিন্ন সংখ্যক cycle নেয় তাহলে:-]

(*) Total clock cycles $= \sum_{i=1}^{n} (CPI_i \times IC_i)$

IC_i → Instruction Count;

CPI_i → Cycle per Instruction
[মানে প্রত্যেক instruction → কতটি cycle নাচ্ছে।]

⊛ Weighted Average CPI

$$\text{Avg. CPI} = \frac{\text{Clock Cycles}}{\text{Total Instruction count}}$$

$$\Rightarrow \text{Avg. CPI} = \sum_{i=1}^{n} \left(\frac{\text{CPI}_i \times \text{Instruction count}_i}{\text{Total Instruction count}} \right)$$

Now Lets Understand Theory

1 add = 3 cycles
1 sub = 2 cycles
1 beq = 4 cycles

We have, 2 add, 2 sub, 1 beq :-

So,

2 add = 2 × 3 = 6
2 sub = 2 × 2 = 4
1 beq = 1 × 4 = 4

Total cycles = 14 cycles

Example:

- Alternative compiled code sequences using instruction in class A, B, C.

Instruction	add	sub	addi
CPI for class	1	2	3
IC in sequence 1	2	1	2
IC in sequence 2	4	1	1

Soln:

(a) In Sequence-1:

2 add = 1 cycle

1 sub = 2 cycle

2 addi = 3 cycle

Total Instruction = 5

$\therefore$ Total cycle $= (2\times1)+(1\times2)+(2\times3)$

$= 10$

$\therefore$ Avg. CPI $= \dfrac{\text{Total cycle}}{\text{Total Instruction}}$

$$\Rightarrow \text{Avg. CPI} = \frac{10}{5} = 2$$

(*) In Sequence-2:

4 add = 1 cycle

1 sub = 2 cycle

1 addi = 3 cycle

Total Instruction = (4 + 1 + 1)

= 6

Total cycle = (4 × 1 + 1 × 2 + 1 × 3)

= 4 + 2 + 3 = 9

$$\therefore \text{Avg. CPI} = \frac{9}{6} = 1.5$$

Power Trend

(*) Clock Rate যত বেশি computer এর performance তত ভালো।

(*) Power অনেক বেশি হলে heat অনেক বেশি হবে। ফলে Computer slow হয়ে যায়।

$$\text{Power} = \text{Capacity load} \times \text{Voltage}^2 \times \text{Frequency}$$

(*) So power fix রাখতে হবে otherwise computer slow হয়ে যাবে or অতিরিক্ত heat হলে computer চলাচল নষ্ট হয়ে যায়।

(*) এখন Computer এর performance বাড়াতে হলে frequency বাড়াতে হবে।

But তা করলে power বেড়ে যায়।
So, power constant রাখতে চাই। এর
জন্য Frequency এমনভাবে বাড়াবো
এবং Voltage এমনভাবে কমাবো
যাতে power constant থাকে।

→ আমরা Performance আরও Better
করতে চাই। But voltage আর কমানো
যায় না। তাই Performance বাড়ানোর
জন্য Single core থেকে Dual core
বা Multicore বানিয়ে Performance
আরো Better করা যায়।

[Multiprocessor দিয়ে এভাবে performance
অনেক বাড়ানো যায় without increasing power.]

Pitfall: Amdahl's Law

[কতটুকু improve করা যায় নাকি না।]

Program এর যে অংশ গুলোকে improve করা যায়।

$$T_{improve} = \frac{T_{affected}}{improvement\ factor} + T_{unaffected}$$

Overall কাজটা করতে যে পরিমান সময় লাগতেছে।

যে time গুলোকে improve করা যায় না।

এটা n time better করতে চলা লাগে n দ্বারা ভাগ করা।

* Improvement factor positive হলে improvement possible.

* Zero or negative or infinity হলে Impossible.

$$\text{Time Improve} = \frac{\text{Total Time}}{\text{How much are want to improve. (n)}}$$

Example: multiply accounts for 80s (80/100) out of 100s.

- How much improvement in multiply performance to get 5x Overall.
- Can't be done!

Soln: n = 5 ~~times faster~~

$$T_{improve} = \frac{T_{total}}{n} = \frac{100s}{5} = 20s$$

$$T_{affected} = 80s$$

$$T_{unaffected} = T_{total} - T_{affected}$$

$$= 100 - 80 = 20s$$

$$\text{So, } T_{improve} = \frac{T_{affected}}{\text{Improvement factor}} + T_{unaffected}$$

$$\Rightarrow 20 = \frac{80}{F.F.} + 20$$

$$\Rightarrow F.F. = 0$$

$\therefore$ So, Improvement Factor = 0

So, Improvement is not possible.