CSE-323
Computer Architecture

Week 2: Computer Performance

Lecturer Afia Anjum
Military Institute of Science and technology

Computer Performance

- What do you understand by computer performance?
- By which you define computer performance?

Time!!!

Computer Performance

- Two basic parameters for measuring computer performance:
  - Response Time/ Execution Time/ CPU Time/ Elapsed Time/ Latency
  - Throughput

Response Time

- It is the time between the start and the completion of an event.
- It is also known as Execution Time/ CPU Time/ Elapsed Time/ Latency.
- Example: Milliseconds for a disk access.

Throughput

- The number of processes that are completed per unit time (day, hour etc.) is called the throughput.
- It is the total amount of work done in a given time.
- It is also known as Bandwidth.
- Example: Megabytes per second for a disk transfer.

Response Time and Throughput

- If you decrease Response Time, Throughput will be increased and vice versa.
- Example: Disk Performance

Performance

- To maximize performance, we obviously need to minimize execution time.
- Performance is inversely related to execution time.
- Performance = $\dfrac{1}{Execution\ Time}$

Relative Performance

- If processor X is n times faster than Y, then we can say,
  - $\text{Performance}_x > \text{Performance}_y$
  - $\text{Performance}_x = \dfrac{1}{Execution\_time_x}$
  - $\text{Performance}_y = \dfrac{1}{Execution\_time_y}$
  - $\dfrac{Performance_x}{Performance_y} = \dfrac{Execution\_timey}{Execution\_time_x} = n$

Relative Performance

Practice problem:

If computer A runs a program in 10 seconds and computer B runs the same program in 20 seconds, how much faster is A than B?

Solution:

$$\frac{Execution\_time_{\mathrm{B}}}{Execution\_time_{\mathrm{A}}} = \frac{20s}{10s} = 2$$

So, A is 2 times faster than B.

CPU Clocking

- Each program is made up of a number of instructions.
- Each instruction takes a number of clock cycles to execute.
- The speed of a computer processor, or CPU, is determined by the **clock cycle**.
- Clock Cycle is the amount of time between two pulses of an oscillator.
- A clock cycle is the basic unit of time to execute one operation
- The clock rate (clock cycles per second in MHz or GHz) is inverse of clock cycle time (clock period),

10^6 10^9

$$CC = \frac{1}{CR}$$

CPU Time

CPU Time/ Execution Time for a Program,

= Number of clock cycles for a program* Clock cycle time

$$= \frac{Number\ of\ clock\ cycles\ for\ a\ program}{Clock\ rate}$$

Performance

How to increase the performance???

- Reduce number of clock cycles

  or,

- Reduce the length of clock cycle time

Performance

Practice problem:

A program runs on computer A with a 2 GHz clock in 10 seconds. What clock rate must a computer B run at to run this program in 6 seconds? Unfortunately, to accomplish this, computer B will require 1.2 times as many clock cycles as computer A to run the program.

Solution:

Computer A:

Clock rate: 2 GHz

Execution time/ CPU time: 10s

Computer B:

Aim for 6s Execution time

No. of clock cycles= 1.2 * No. of clock cycles of A

Clock rate= ???

Performance

Solution:

$$\text{Execution\_time}_B = \frac{\text{No. of clock cycles}_B}{\text{Clock rate}_B}$$

$$\text{Clock rate}_B = \frac{\text{No. of clock cycles}_B}{\text{Execution\_time}_B} = \frac{1.2 \text{ x No. of clock cycles}_A}{\text{Execution\_time}_B}$$

$$\text{No. of clock cycles}_A = \text{Execution\_time}_A \text{ x Clock rate}_A$$

$$= 10\text{s x } (2 \text{ x } 10^9)\text{cycles/s}$$

$$= 20 \text{ x } 10^9 \text{ cycles}$$

$$\text{Clock rate}_B = \frac{1.2\text{x No. of clock cycles}_A}{\text{Execution\_time}_B} = \frac{1.2 \text{ x } 20 \text{ x } 10^9 \text{ cycles}}{6\text{s}}$$

$$= \mathbf{4\ Ghz}$$

Cycles Per Instruction (CPI)

- A single machine instruction may take one or more CPU cycles to complete, which is termed as the Cycles Per Instruction (CPI)
- Using CPI, we can redefine previous formulas for a single program:

$$\text{CPI} = \frac{\text{Clock Cycles}}{\text{No. of Instructions}}$$

Clock Cycles = No. of Instructions x Cycles Per Instruction(CPI)

Execution Time = No. of Instructions x CPI x Clock Cycle Time

$$= \frac{\text{No. of Instructions x CPI}}{\text{Clock rate}}$$

Cycles Per Instruction (CPI)

Practice problem:

Computers A and B implement the same no of Computer Machine Instructions. Computer A has a clock cycle time of 250 ps and an effective CPI of 2.0 for some program and computer B has a clock cycle time of 500 ps and an effective CPI of 1.2 for the same program. Which computer is faster and by how much?

Solution:

Each computer executes the same number of instructions, I, so,

$\text{Execution\_time}_A = I \times 2 \times 250 \text{ ps} = I \times 500 \text{ ps}$

$\text{Execution\_time}_B = I \times 1.2 \times 500 \text{ ps} = I \times 600 \text{ ps}$

It is clear that A is faster than B.

Now, $\dfrac{\text{Performance}_A}{\text{Performnace}_B} = \dfrac{\text{Execution\_time}_B}{\text{Execution\_time}_A} = \dfrac{I \times 600 \text{ ps}}{I \times 500 \text{ ps}} = 1.2$

So, A is 1.2 times faster than B.

Cycles Per Instruction (CPI)

Practice Problem:

A given application written in Java runs 15 seconds on a desktop processor.

A new Java compiler is released that requires only 0.6 as many instructions as the old compiler.

Unfortunately, it requires 1.1 as many CPI as the old compiler.

How fast can we expect the application to run using this new compiler? (Assume that, Clock Cycle Time of both compiler is same)

Cycles Per Instruction (CPI)

Solution:

Old Compiler:

$$Execution\_time_{Old} = No\_of\_Ins_{Old} * CPI_{Old} * CC\_time$$

$$CC\_time = \frac{Execution\_time_{Old}}{No\_of\_Ins_{Old} * CPI_{Old}}$$

New Compiler:

$$Execution\_time_{New} = (0.6 * No\_of\_Ins_{Old}) * (1.1 * CPI_{Old}) * CC\_time$$

$$= \frac{(0.6 * No\_of\_Ins_{Old}) * (1.1 * CPI_{Old}) * 15s}{No\_of\_Ins_{Old} * CPI_{Old}}$$

$$= 9.9s$$

[Now determine how fast by yourself]

Cycles Per Instruction (CPI)

Practice problem:

- A compiler designer is trying to decide between two code sequences for a particular machine.
- Based on the hardware implementation, there are three different classes of instructions: Class A, Class B, and Class C,

	CPI for each instruction class		
	A	B	C
CPI	1	2	3

Code sequence	Instruction counts for each instruction class		
	A	B	C
1	2	1	2
2	4	1	1

- Which code sequence has the most instructions? Which sequence will be faster? Assume that, CC Time = 1s. What is the CPI for each sequence?

Cycles Per Instruction (CPI)

Solution:

Which code sequence has the most instructions?

Sequence 1 executes 2+1+2= 5 instructions

Sequence 2 executes 4+1+1= 6 instructions

Therefore, sequence 1 executes fewer instructions.

Which code sequence will be faster?

$\text{Execution\_time}_1 = (\sum_{i=1}^{n}(No\_of\_instructions_i * CPI_i)) * \text{CC time}$

$=((2*1)+(1*2)+(2*3))*1s$

$=10s$

$\text{Execution\_time}_2 = (\sum_{i=1}^{n}(No\_of\_instructions_i * CPI_i)) * \text{CC time}$

$=((4*1)+(1*2)+(1*3))*1s$

=9s (So Sequence 2 is faster)

Cycles Per Instruction (CPI)

Solution:

What is the CPI for each code sequence?

$$\text{CPI}_1 = \frac{\sum_{i=1}^{n}(No\_of\_instructions_{\text{i}} * CPI_{\text{i}})}{\sum_{i=1}^{n}(No\_of\_instructions_{\text{i}})}$$

$$= \frac{(2*1)+(1*2)+(2*3)}{2+1+2} = \frac{10}{5} = 2$$

$$\text{CPI}_2 = \frac{\sum_{i=1}^{n}(No\_of\_instructions_{\text{i}} * CPI_{\text{i}})}{\sum_{i=1}^{n}(No\_of\_instructions_{\text{i}})}$$

$$= \frac{(4*1)+(1*2)+(1*3)}{4+1+1} = \frac{9}{6} = 1.5$$

Benchmark

- It is a test that measures the performance of hardware, software, or computer.
- The higher the value of the results of benchmark tests, the faster the component, software, or overall computer is.
- **The Standard Performance Evaluation Corporation (SPEC)** is a non-profit corporation trusted by many for performing tests.
- It was formed to establish, maintain and endorse standardized benchmarks and tools to evaluate performance.

“Respect your parents, they passed school without Google !”