Performance

To measure success in computer architecture, it is necessary to compare alternative designs. A better computer has better performance, but what is performance?

Plane	Passengers	Range	Speed	Speed*Passengers
B 777	375	4630	610	228,750
B 747	470	4150	610	286,700
Concorde	132	4000	1350	178,200
DC-8-50	146	8720	544	79,424

(c) Copyright Larry Snyder 1998

Defining Performance

- $\text{Performance}_M = 1/\text{Execution Time}_M$

  $\text{Performance}_{M1} > \text{Performance}_{M2}$

  $1/\text{Execution Time}_{M1} > 1/\text{Execution Time}_{M2}$

  $\text{Execution Time}_{M2} > \text{Execution Time}_{M1}$

- Machine M1 is *n* times faster than M2

$$\frac{\text{Performance}_{M1}}{\text{Performance}_{M2}} = n = \frac{\text{Execution Time}_{M2}}{\text{Execution Time}_{M1}}$$

(c) Copyright Larry Snyder 1998

Measuring Performance

- Time = Elapsed time, Response Time, Wall Clock Time
- CPU Execution Time (CPU Time) = time CPU applies to program
  - User CPU Time = time CPU spends on user code
  - System CPU Time = time CPU spends in system for user
- CPU Performance = User CPU time
- System Performance = Elapsed time on unloded system
- Hardware Clock
  - Many terms: clock cycles, clock periods, ticks, clock ticks, cycles
  - Clock period can be cycle time or clock rate
  - Cycle time (sec) is inverse of clock rate (Hz)

(c) Copyright Larry Snyder 1998

Relating Metrics

- Execution time is defined by instructions and their execution time
- CPU execution time = CPU clock cycles $\times$ cycle time
- Equivalently,

  CPU exectuion time = CPU clock cycles/clock rate
- Relating execution time to instruction execution:

  CPI = (Average) clock cycles per instruction

>> Basic Formula of Computer Architecture <<

CPU time = Instruction count x CPI x cycle time

- Equivalently, CPU time = $\frac{\text{Instruction count} \times \text{CPI}}{\text{Clock rate}}$

These equations relate three fundamental quantities

(c) Copyright Larry Snyder 1998

Computing CPI, An Example

Inst Class	Cycles
A	1
B	2
C	3

- Consider two code sequences:
  - A A B C C = 5 instructions
  - A A A A B C = 6 instructions
- Compute CPU clock cycles
  - CPU clocks = 2x1 + 1x2 + 2x3 = 2 + 2 + 6 = 10
  - CPU clocks = 4x1 + 1x2 + 1x3 = 4 + 2 + 3 = 9
- CPI = CPU clock cycles/Instruction count
  - CPI_1 = 10/5 = 2
  - CPI_2 = 9/6 = 1.5

(c) Copyright Larry Snyder 1998

MIPS Rate

- MIPS = Instr count/Execution time x 10^6
  = Instr count/CPU clocks x cycle time x 10^6

$$= \frac{\text{Instr count x clock rate}}{\text{Instr count x CPI x } 10^6} = \frac{\text{Clock rate}}{\text{CPI x } 10^6}$$

- Execution time = $\dfrac{\text{Instr count x CPI}}{\dfrac{\text{Clock rate x } 10^6}{\text{CPI x } 10^6}}$
- $= \dfrac{\text{Instruction count}}{\text{MIPS x } 10^6}$
- Three Problems:
  - Gives instruction rate, but is independent of instr set
  - MIPS varies between programs on same computer
  - MIPS can vary inversely with performance

(c) Copyright Larry Snyder 1998

Computer MIPS Rate

- Two code sequences: A^5 B C, A^{10} B C;
- Compute CPI for each sequence with 100MHz Clock:
  - $CPI_1 = \frac{(5x1+1x2+1x3) \times 10^6}{(5+1+1) \times 10^6} = \frac{10}{7} = 1.43$
  - $CPI_2 = \frac{(10x1+1x2+1x3) \times 10^6}{(10+1+1) \times 10^6} = \frac{15}{12} = 1.25$
  - $MIPS_1 = 100MHz/1.43x10^6 = 69.9$
  - $MIPS_2 = 100MHz/1.25x10^6 = 80.0$
- Execution time = Instruction count x CPI/clock rate
  - $CPU\ time_1 = ((5+1+1)x10^6x1.43)/100x10^6 = 7x1.43/100 = 0.10sec$
  - $CPU\ time_2 = ((10+1+1)x10^6x1.25)/100x10^6=12x1.25/100=0.15sec$

(c) Copyright Larry Snyder 1998

MFLOPS

- MFLOPS = Number floats/Execution time x 10^6
- Program dependent
- More consistent across machines than MIPS
- Not all floating point operations have the same CPI
- "Peak MFLOPS" most egregious

(c) Copyright Larry Snyder 1998