Basic Performance Eqn: $T = n \cdot CPI \cdot (\Delta T/\text{cycle})$

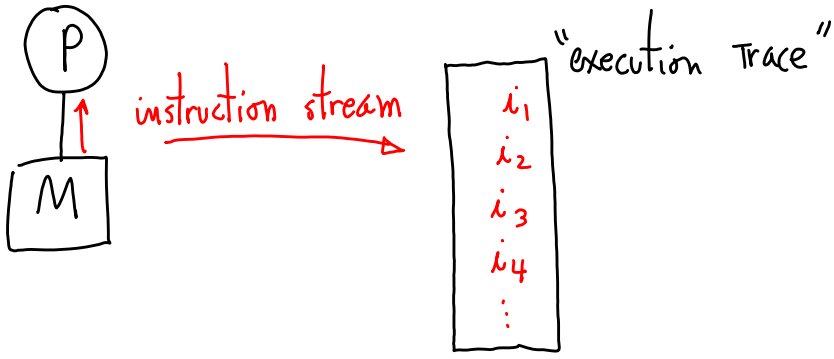
- T: Total # instructions
- CPI: #cycles/instruction
- $(\Delta T/\text{cycle})$: time/cycle

$(\Delta T/\text{cycle})$ = Cycle-Time

$\left(\frac{\text{cycles}}{\Delta T}\right)$ = Clock-Rate

Where did n come from?
How do we measure it?

Program P (machine code) executes on machine M



Actually, not quite so good:
--- Want to see sequence into IR
Instruction caching ==> not one memory access per instruction execution

---

Consider,

Machine	CPI_{add}	CPI_{BR}	CR (cycles/sec)
M_1	1	3	1.5 GHz
M_2	2	2	2 GHz

Program Trace

f_{add} = 70% ADD instructions

f_{BR} = 30% BR instructions

Find $S_{1\text{-}2} = {}^{V_1}/_{V_2} = \frac{W/T_1}{W/T_2} = {}^{T_2}/_{T_1}$ Which is faster?

$T_1 = T_{add} + T_{BR} = n_{add} \cdot CPI_{add} \cdot \text{cycle-Time} + n_{BR} \cdot CPI_{BR} \cdot \text{cycle-time}$

cycle-time = $1/CR$

$$\frac{T_2}{T_1} = \frac{\left(\frac{n_{add}}{n}\right) CPI_{add-2} / CR_2 + \left(\frac{n_{BR}}{n}\right) CPI_{BR-2} / CR_2}{\left(\frac{n_{add}}{n}\right) CPI_{add-1} / CR_1 + \left(\frac{n_{BR}}{n}\right) CPI_{BR-1} / CR_1}$$

$$\frac{n_{add}}{n} = f_{add}$$

$$S_{1-2} = \frac{(0.7)\,2 + (0.3)\,2}{(0.7)\,1 + (0.3)\,3}\left(\frac{CR_1}{CR_2}\right) = \frac{200}{160}\left(\frac{1.5\ GHz}{2\ GHz}\right)$$

$$= \frac{5}{4}\left(\frac{3}{4}\right) = 15/16$$

$$S_{2-1} = 1/S_{1-2} = 16/15 = 1\,1/15 \approx 7\%\ \text{faster w/ 33\% CR}$$

message?
M1 : target common case at expense of others (ADD vs. BR) and at expense of CR
M2 : break ADD into two steps ==> increased CR, benefits both ADD and BR

find

$$\overline{CPI_1} = \frac{n_{cycles}}{n_{instructions}} = \frac{n_{add\text{-}cycles} + n_{BR\text{-}cycles}}{n} = \left(\frac{n_{add}}{n}\right)\cdot CPI_{add} + \left(\frac{n_{BR}}{n}\right)\cdot CPI_{BR}$$

$$= (0.7)\,1 + (0.3)\,3$$

$$= 0.7 + 0.9 = 1.6$$

$$\overline{CPI_2} = (0.7)\,2 + (0.3)\,2$$

$$= 1.4 + 0.6 = 2.0$$

Message?

--- Avg CPI is important
--- Determined by job mix

What else?

--- Cache effects, instruction order, ISA

swap job mix?

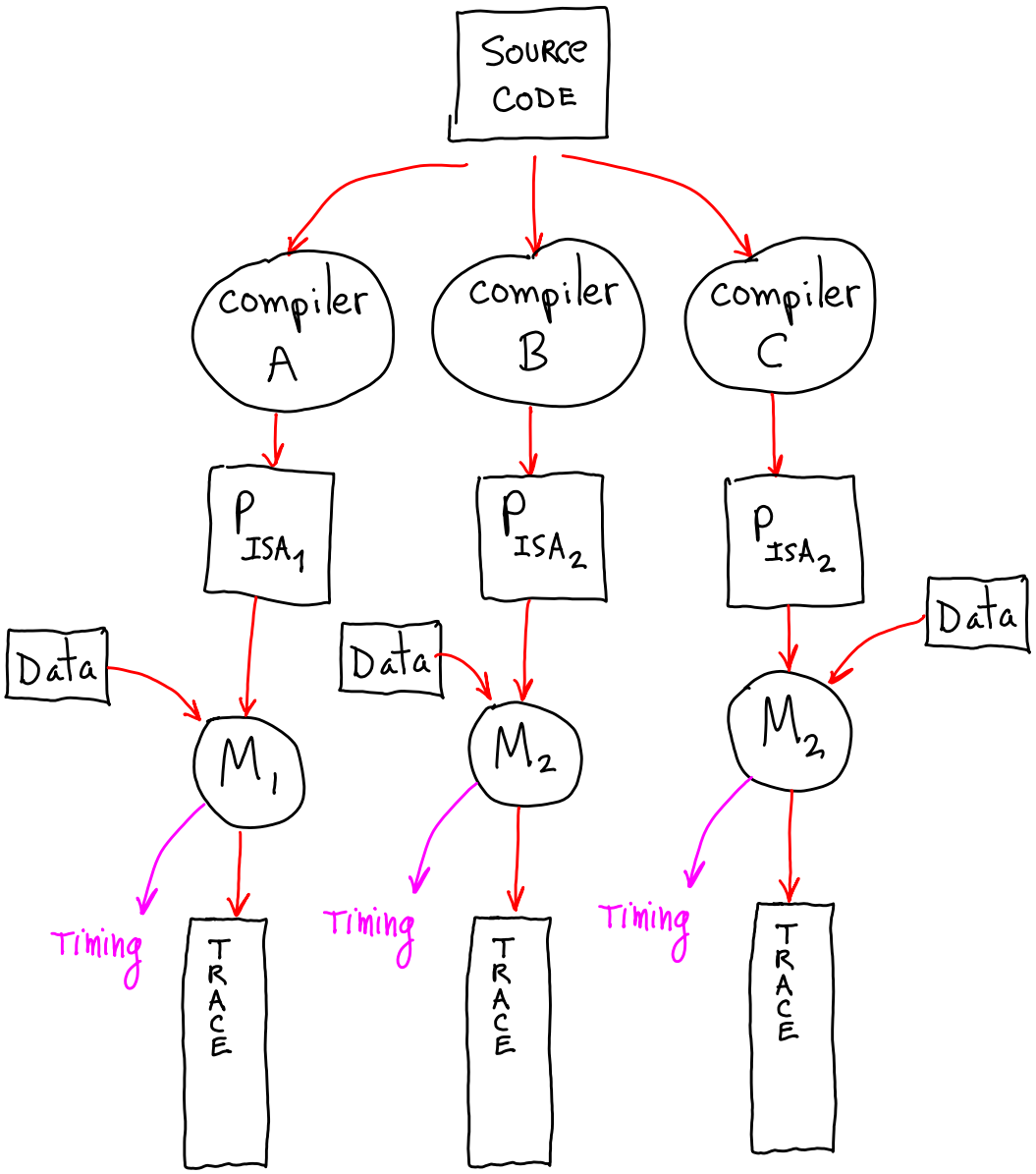
$$(0.3)\,1 + (0.7)\,3 = 2.4$$

$$(0.3)\,2 + (0.7)\,2 = 2.0$$

How do we measure cycles per instruction?

1. Count execution states — Ok for simple machines, e.g., LC3
2. Hardware counters — doesn't always gives you what you want.
3. Simulation — pretty good, but timing-accurate simulation is difficult
4. Execute and time it — Try to see what only ADDs require, e.g.
   ==> Make a guess at CPI per class

   Measure overall time ==> #cycles
   Try varying % ADDS, see effect



Things that make a difference:

--- source Program

--- Compiler

--- ISA

--- micro-architecture

--- Data

--- Execution environment

	CPI_{avg}	CR
M_1	1.5	2 GHz
M_2	1.0	1.5 GHz
M_3	2.5	3 GHz

Reduce execution time by 30% by increasing CR → $CPI_{new} = 1.2\ CPI_{old}$ penalty

find new CR.

$$T_1 = n(1.5)/2GHz = \frac{3n}{4}\ ns$$

$$T_2 = n(1.0)/\frac{3}{2}GHz = \frac{2n}{3}\ ns$$

$$T_3 = n(2.5)/3GHz = \frac{5n}{6}\ ns$$

$$\frac{V_{new}}{V_{old}} = 1.3 = \frac{T_{old}}{T_{new}}$$

$$T_{new} = T_{old}/1.3$$

$$T_1^{new} = (77\%)T_1 = (0.77)\frac{3n}{4}\ ns = n\ CPI_1^{new}/CR_1^{new}$$

$$= n(1.2 \cdot CPI_1)/CR_1^{new}$$

$$\frac{21}{40}n\ (ns) = n(1.2)(1.5)/CR'_{new} \Rightarrow CR_{new} = \left(\frac{40}{21}\right)\left(\frac{12}{10}\right)\left(\frac{15}{10}\right)\left(\frac{1}{ns}\right)$$

$$= \left(\frac{48}{21}\right)\left(\frac{3}{2}\right)GHz = \frac{24}{7}GHz$$

$$\approx 3.5\ GHz$$

find fastest Machine

	CR	CPI_A	CPI_B	CPI_C	CPI_D	CPI_E
M1	1	1	2	3	4	3
M2	1.5	2	2	2	4	4
M3	1	1	1	2	3	2
M4	1.5	1	2	3	4	3

Given, Trace $= (i_1, i_2, i_3, \dots, i_N)$

* (class A instructions) $= 2n$
* (class B instructions) $= n$
* (class C instructions) $= n$
* (class D instructions) $= n$
* (class E instructions) $= n$

$N = 6n$

$$\overline{CPI} = \frac{n_A CPI_A + n_B CPI_B + \cdots + n_E CPI_E}{N}$$

$$= \frac{2n\, CPI_A + n\, CPI_B + \cdots + n\, CPI_E}{6n}$$

$$= (2\, CPI_A + CPI_B + \cdots + CPI_E)/6$$

$$S_{1-2} = V_1/V_2 = \frac{W/T_1}{W/T_2} = \frac{T_2}{T_1} = \frac{N\ \overline{CPI_2}\ (1/CR_2)}{N\ \overline{CPI_1}\ (1/CR_1)} = \left(\frac{\overline{CPI_2}}{\overline{CPI_1}}\right)\frac{CR_1}{CR_2}$$

$$= \frac{CPI_A^{(2)} + \left(\sum_i CPI_i^{(2)}\right)}{CPI_A^{(1)} + \left(\sum_i CPI_i^{(1)}\right)}\ \frac{CR_1}{CR_2}$$

$$= \frac{2 + (2\ \ 2\ \ 2\ \ 4\ \ 4)}{1 + (1\ \ 2\ \ 3\ \ 4\ \ 3)}\left(\frac{1}{1.5}\right) = \frac{16}{14}\left(\frac{2}{3}\right) = 16/21$$

$$= \sqrt[n]{\prod^{n} \left(S_{B-A_i}\right)}$$

$$\overline{S_{A-B}} = \frac{1}{2^{1/4}}$$

$$e^{\left(\frac{1}{n}\sum \ln(T_i)\right)} \qquad \text{stdev} \sqrt{\frac{\sum^{n}(X_i - \bar{X})^2}{A}}$$

$$\text{stdev} = e^{\sqrt{\frac{1}{n}\left(\sum(\ln(T_i) - \ln(\bar{T}))^2\right.}}$$

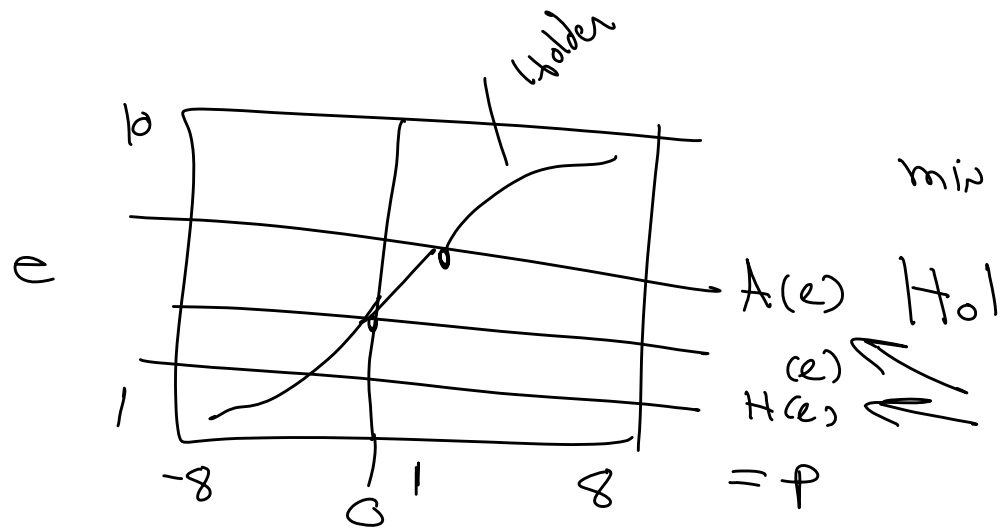
$$\text{stdev}_A > \text{stdev}_B \Rightarrow ?$$

Other comparisons?

$$M(e) = M(P(e))$$

$$cM(e) = M(ce)$$

$$\min \le M(e) \le \max$$



$$\text{Holder} = \left(\frac{1}{n}\sum a_i^p\right)^{\frac{1}{p}}$$

$$f(m) = \frac{f(e_1) + f(e_2) + \cdots + f(e_n)}{n}$$

$$H_k \Rightarrow f(x) = x^k$$

$$H_1(e) \Rightarrow m^1 = \frac{e_1^1 + e_2^1 + \cdots + e_n^1}{n} = AM$$

$$H_{(-1)} \Rightarrow m^{-1} = \frac{e_1^{-1} + e_2^{-1} + \cdots + e_n^{-1}}{n}$$

$$\text{or} \quad m = \frac{n}{\sum_i 1/e_i} = HM$$

$$\lim_{k\to\infty}\left(\quad m^k = \frac{e_1^k + e_2^k + \cdots + e_n^k}{n} \right) \Rightarrow \log(m) = \frac{\log(e_1) + \cdots + \log(e_n)}{n}$$

$$m = \left(\frac{e_1^k + e_2^k + \cdots e_n^k}{n}\right)^{1/k}$$

$$m = \exp\left(\log(e_1) + \log(e_2) + \cdots + \log(e_n)\right)^{\frac{1}{n}}$$

$$= (e_1 \cdot e_2 \cdot \cdots e_n)^{1/n}$$