L4 comparch

- Throughput = No.of jobs completed / time
  it is an average
- improving Throughput doesn't imprpve response time i.e. the time required per instruction. Eg. 2 cores of 1.66 ghz improves throughput but needn't necessarily improve Response time
- if response time is imporved however throughput is improved

Performance - I: speedup

$$\frac{perf\ X}{perf\ Y} = \frac{exec\ time\ y}{exec\ time\ x}$$

Exec time is user CPU time. (time taken to run a program, not system program)

2 exec time

- $\frac{Seconds}{program} = \frac{cycles}{program} \times \frac{seconds}{cycle}$
- $CPU\ Exec\ time = CPU\ clock\ cycles * clock\ cycle\ time$
- $Exec\ time\ = Clock\ cycle\ time\ * number\ of\ instructions\ * avg.\ CPI$
  (CPI = Cycles per instruction)
  - eg. first instruction takes 2, second 4 etc.
- to improve performance, we can reduce number of cycles per program
  1. increase number of resources
  2. pipeline (reduces average number)
  3. making improvement in other hardware such as control logic etc.
  4. We can't increase frequency beyond a certain extent, as leakage current will increase

How many cycles per program

- Operations can take > 1 cycle
- Different operations can take different cycles

Question

4 * 10^9 *1.5 / 1Ghz = 6 seconds
2 * 10^9 *6 / 1.5 = 8 seconds

10 ns * 2.0 CPI = 20ns per inst
20 ns * 1.2 CPI = 24ns per instruction

Benchmarks

- expected workload
- use realworld application
    - Note that small benchmarks can be abused!
- Each vendor releases a SPEC rating (System Performance Evaluation Corporation)
    - 12 integer and 17 floating point applications as of 2006
    - SPEC rating specifies how much faster a system is compared to a baseline machine
    - Incorporates behavior of these 29 programs, not your fav. program </3