Example

- Our favorite program runs in 10 seconds on computer A, which has a 400 Mhz. clock. We are trying to help a computer designer build a new machine B, that will run this program in 6 seconds. The designer can use new (or perhaps more expensive) technology to substantially increase the clock rate, but has informed us that this increase will affect the rest of the CPU design, causing machine B to require 1.2 times as many clock cycles as machine A for the same program. What clock rate should we tell the designer to target?

$$\frac{\text{seconds}}{\text{program}} = \frac{\text{cycles}}{\text{program}} \times \frac{\text{seconds}}{\text{cycle}}$$

For program A: 10 seconds = $\text{Cycles}_A \times 1/400\text{MHz}$

For program B: 6 seconds = $\text{Cycles}_B \times 1/\text{clock rate}_B$

$\text{Cycles}_B = 1.2\ \text{Cycles}_A$

$\text{Clock rate}_B = 800\text{MHz}$

Now that we understand cycles

- A given program will require
  - some number of instructions (machine instructions)
  - some number of cycles
  - some number of seconds
- We have a vocabulary that relates these quantities:
  - cycle time (seconds per cycle)
  - clock rate (cycles per second)
  - CPI (cycles per instruction)
    a floating point intensive application might have a higher CPI
  - MIPS (millions of instructions per second)
    this would be higher for a program using simple instructions

Another Way to Compute CPU Time

$$\text{CPU Time (or, Execution Time)} = \frac{\text{\# of instructions}}{\text{program}} \times \frac{\text{\# of cycles}}{\text{instruction}} \times \frac{\text{\# of seconds}}{\text{cycle}}$$

$$= \text{instruction count} \times \text{CPI} \times \text{cycle time}$$

$$= \text{instruction count} \times \text{CPI} \times \frac{1}{\text{clock rate}}$$

Performance

- Performance is determined by execution time
- Do any of the following variables alone equal performance?
  - # of cycles to execute program?
  - # of instructions in program?
  - # of cycles per second?
  - average # of cycles per instruction (CPI)?
  - average # of instructions per second?

- Common pitfall: thinking one of the variables is indicative of performance when it really isn't.

CPI Example

- Suppose we have two implementations of the same instruction set architecture (ISA).

  For some program P,

  Machine A has a clock cycle time of 10 ns. and a CPI of 2.0
  Machine B has a clock cycle time of 20 ns. and a CPI of 1.2

  What machine is faster for this program, and by how much?

  CPU $time_A$ = IC × CPI × cycle time = IC × 2.0 × 10ns = 20 ×IC ns
  CPU $time_B$ = IC × 1.2 × 20ns = 24 ×IC ns
  So, A is 1.2 (=24/20) times faster than B

- If two machines have the same ISA which of our quantities (e.g., clock rate, CPI, execution time, # of instructions, MIPS) will always be identical?

of Instructions Example

- 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, and they require one, two, and three cycles (respectively).

  The first code sequence has 5 instructions: 2 of A, 1 of B, and 2 of C
  The second sequence has 6 instructions: 4 of A, 1 of B, and 1 of C.

  Which sequence will be faster? How much? (assume CPU starts execute the 2nd instruction after the 1st one completes)
  What is the CPI for each sequence?

$\# \text{ of cycles}_1 = 2 \times 1 + 1 \times 2 + 2 \times 3 = 10$

$\# \text{ of cycles}_2 = 4 \times 1 + 1 \times 2 + 1 \times 3 = 9$ So, sequence 2 is 1.1 times faster

$CPI_1 = 10 / 5 = 2$

$CPI_2 = 9 / 6 = 1.5$

MIPS Example

- Two different compilers are being tested for a 100 MHz. machine with three different classes of instructions: Class A, Class B, and Class C, which require one, two, and three cycles (respectively). Both compilers are used to produce code for a large piece of software.

  The first compiler's code uses 5 million Class A instructions, 1 million Class B instructions, and 1 million Class C instructions.

  The second compiler's code uses 10 million Class A instructions, 1 million Class B instructions, and 1 million Class C instructions.

- Which sequence will be faster according to MIPS?
- Which sequence will be faster according to execution time?

$\#\text{ of instruction}_1 = 5M + 1M + 1M = 7M$, $\#\text{ of instruction}_2 = 10M + 1M + 1M = 12M$

$\#\text{ of cycles}_1 = 5M \times 1 + 1M \times 2 + 1M \times 3 = 10M\text{cycles} = 0.1\text{ seconds}$

$\#\text{ of cycles}_2 = 10M \times 1 + 1M \times 2 + 1M \times 3 = 15M\text{ cycles} = 0.15\text{ seconds}$

So, $MIPS_1 = 7M/0.1 = 70MIPS$, $MIPS_2 = 12M/0.15 = 80MIPS > MIPS_1$