

## **Computer Performance**

**1-29-2003**

---

---

---

---

---

---

---

---

## **Opening Discussion**

- What did we talk about last class?
- When you buy a computer, what is the most important thing you look for?
- The future is ... Minority Report? Just like today? Something in between?
- NVIDIA is hiring. TopCoder has information. (The FBI is on there too.)

---

---

---

---

---

---

---

---

## **Definition of Performance**

- For many computer buyers, what matters most about the computer they buy is how well it performs. Basically, they want it to be fast. This is a very hard thing to measure though. It would be ideal if we could report a single number and have that give us the relative performance of a machine.
- Reality is price to performance ratio, but that requires knowing performance.

---

---

---

---

---

---

---

---

## Measuring Time

- The real metric for measuring the performance of machine is time. How long does it take to do something or how much can it do in a certain period of time.
- Wall-clock time: This is easy to measure, but not all that accurate
- CPU time
  - User vs. System
- The time command will measure for you.

---

---

---

---

---

---

---

---

## Clock Cycles/Clock Speed

- This is the metric you see most of the time. Basically, it is just a comparison of how many cycles the processor goes through each second.
- It is really a very poor one because different machines get different amounts of work done in one clock cycle.
- Different marketing departments push this at different times.

---

---

---

---

---

---

---

---

## Not All Instruction Created Equal

- Even on a single processor, different type of instructions can take different numbers of clock cycles to complete. As a result, the performance of a given program can depend on what types of instructions it executes.
- Divided and memory loads are some of the worst instructions, performance wise, on a modern processor.

---

---

---

---

---

---

---

---

## Clock Cycles per Instruction

- What we'd really love to have is something of a general average of how long each instruction takes to execute on the machine in question, given a "typical program".
- To measure this you have to use a profiler of a simulator to count how many instructions execute for that program with the supplied input, the clock speed, and the execution time.

---

---

---

---

---

---

---

---

## Breaking Down Time

- We can express execution time as the product of three values.

$$Time = \frac{Instructions}{Program} \times \frac{ClockCycles}{Instruction} \times \frac{Seconds}{ClockCycle}$$

- While this is quite elementary, it helps us to see the key components that go into the overall performance of a machine.

---

---

---

---

---

---

---

---

## Role of Compilers

- Good compilers can be absolutely critical for the performance of a certain platform. For example, Intel compilers will typically produce code that is ~30% faster than what gcc makes.
- Properly utilizing the instruction set of a machine is important.
- MIPS and IA-64 put more pressure on compilers for performance.

---

---

---

---

---

---

---

---

## Software Benchmarks

- You want to use real world programs. That varies a lot between different users of course. What you don't want is small programs that only do certain very specific types of operations.
- There are a number of good benchmarks for windows boxes that use normal apps. For general processing though you run into the problem of porting.

---

---

---

---

---

---

---

---

## Compiling Results

- We are really into having a single number that tells us how fast a computer is. Possibly 2 or 3 for different components.
- If we run several different programs, how can we get a single value that accurately compares two computers based on that?
- Your book proposed weighted averages. That requires that you know something about the use behaviors.

---

---

---

---

---

---

---

---

## SPEC Benchmarks

- The most widely used general benchmark is the SPEC benchmark suite.
  - <http://www.spec.org>
- Since the book came out they retired the SPEC CPU95 tests and went to the SPEC CPU2000 benchmarks.
- SPEC also has a bunch of other benchmarks for different things.

---

---

---

---

---

---

---

---

## Minute Essay

- This chapter is a bit more math intensive, but it's just basic algebra. Hopefully this isn't a problem, but if it is let me know.
- If you look at the schedule you might notice that shortened our time on chapter 2 and increased the time on chapter 4. I didn't move quizzes or assignments though.
- Assignment #1 is due next class.

---

---

---

---

---

---

---

---