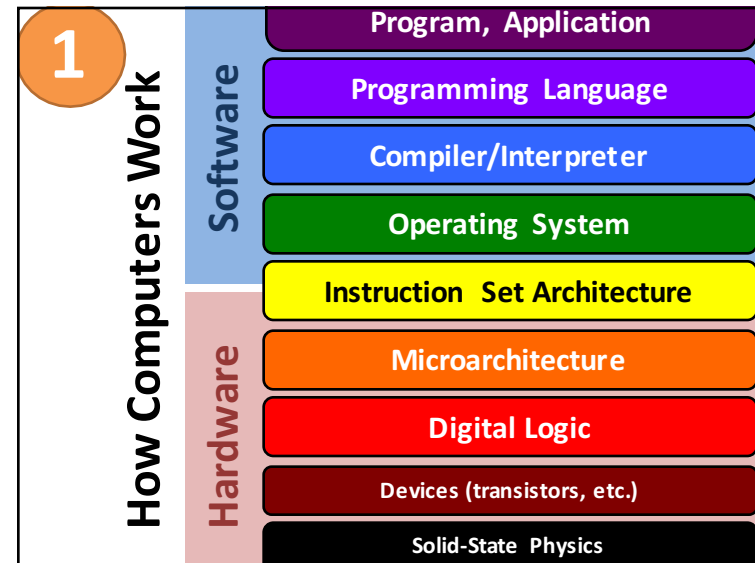


CS 240: Big Ideas, Human Impacts



2 Programming Skills

Few of you will build new HW, OS, compiler, but...

1. Effective programmers understand their tools and systems.
2. The skills and ideas you learn here apply everywhere.

Reason about computational models, translation

Assess costs and limits of representations

Debug for correctness and performance (with tools to help)

A little concurrency

3 Big Ideas in CS, Systems, and beyond

Abstraction

Do not start every project with transistors.
Abstraction is beautiful and empowering,
but real abstractions have leaks and wrinkles.

Translation

Between layers of abstraction.
Structured computation.

Representation

No representation without taxation.
Representations have costs.

Performance

Memory: clever, imperfect abstraction.
Tiny code changes, huge impact.

Security + Reliability



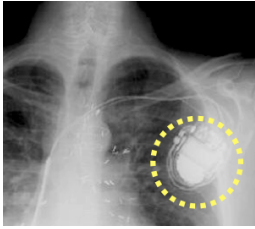
Trickiest exploits & errors
involve multiple layers, even hardware!

These things matter more every day.

How to Detect Exploits of the GHOST Buffer Overflow Vulnerability

Wednesday, February 11, 2016 Swati Khandelwal

8-1 | 75 | | | 593 | | 250 | | 5

DETECTING GHOST VULNERABILITY

The *GHOST vulnerability* is a buffer overflow condition that can be easily exploited remotely, which makes it extremely dangerous. This vulnerability is named after the *GetHOS* function involved in the exploit.

The New York Times Business

A Heart Device Is Found Vulnerable to Hacker Attack

By BARBARA J. FEDER
Published March 12, 2015

To the long list of objects vulnerable to attack by computer hackers, add the human heart.

The threat seems largely theoretical. But a team of computer security researchers plans to report Wednesday that it had been able to gain wireless access to a combination heart defibrillator and pacemaker.

Ariane 5 Rocket, 1996

Exploded due to cast of 64-bit floating-point number to 16-bit signed number. **Overflow.**



1998


Mars Climate Orbiter

Disintegrated due to mismatched units in Lockheed-Martin / NASA software components.



Toyota "Unintended Acceleration Events"

Oklahoma jury:
"Spaghetti Code" = "reckless disregard"



>10,000 global variables
81,514 violations of MISRA-C coding rules
Expect 3 minor bugs + 1 major bug per 30 violations

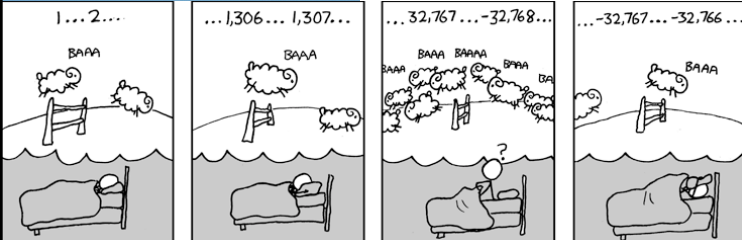
Task/process monitoring failed to monitor tasks/processes
Memory corruption

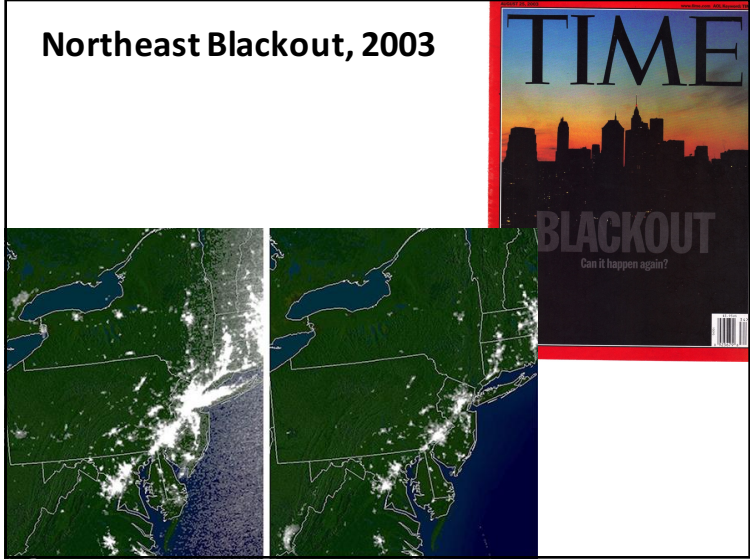
(Wait, it was written in C?!?!?)

<http://www.safetyresearch.net/blog/articles/toyota-unintended-acceleration-and-big-bowl-%E2%80%99Cspaghetti%E2%80%99D-code>

... a **Model 787 airplane** that has been powered continuously for 248 days can lose all alternating current (AC) electrical power due to the generator control units (GCUs) simultaneously going into failsafe mode ... This condition is caused by a **software counter** internal to the GCUs that will **overflow** after **248 days** of continuous power. We are issuing this AD to prevent loss of all AC electrical power, which could result in **loss of control of the airplane.** --FAA, April 2015

<https://xkcd.com/571/>





Things you could improve in the future...

- Reliability
- Security
- Energy-efficiency and manufacturing impacts

Apply computing to improve _____ ... it's 240 at some level!

...

Just a few of the impacts we usually don't see:
http://www.nytimes.com/2015/06/07/magazine/making-and-unmaking-the-digital-world.html?_r=0
What a simple phone can do for people: <https://opendatakit.org/about/deployments/>

