# Programming Abstractions

CS106B

Cynthia Bailey Lee
Julie Zelenski

## Today’s Topics

1. **Final exam discussion**
   - Details/logistics, topics, sources for practice problems
2. **Quarter wrap-up**
   - Putting it all together: what have we accomplished together this quarter?
3. **What next?**
   - Options for continuing your passion for CS after this quarter is done
   - Preview of CS107: security exploits in C/C++

## End of Quarter Info

## End of Quarter Details

- Week 10 assignment schedule:
  - Assignment 7 is out now and due 12/1
  - Huffman Coding!

- Week 10 lecture schedule:
  - Monday: quarter wrap-up lecture
  - Wednesday: ethics roundtable discussion
  - Friday: no official lecture, we’ll have something halfway between a lecture and office hours style time for final exam review

## Final Exam Info

## Final Exam Details

- **Monday, December 6th**
- The testing period will **between 12:01am PST and 11:59pm PST.**
- Unless you've been told otherwise, you will have **3 hours** to take the final.
- It will be administered on Gradescope.
- The exam is open-book, open-notes, open-internet, open-compiler.
- The exam is strictly **closed to assistance from other people** or Q&A forums of any kind (e.g., StackOverflow). See the written exam policies on the course website for more information. Violation of any aspect of this policy will be reported to the Office of Community Standards as an Honor Code violation.

## Final Exam Topics

- ADTs
- Recursion
- Backtracking
- Big-O analysis
- Pointers and dynamic memory, linked lists
- Classes and objects
- Binary heaps
- Binary search trees (BSTs)
- Tree traversals
- Light coverage: Sorting algorithms, Graphs, Hashing

## Final Exam Study Strategy

- Don’t memorize things—either write it in notes*, or learn the concept
  - If you’ve got flash cards, you’re approaching this with the wrong mindset
  - No big multiple choice/true-false section where memorized facts would be tested
  - * even though the exam is open-book, it would be a useful exercise to make a 1-pager notes sheet
- **Read the book** (but **<u>only</u>** in a targeted way)
  - Computer science is about creating things, so do some practice problems
  - Re-do questions from lecture, do old section problems, do the practice exam
  - Look at lecture videos or book **<u>as needed</u>** for review of things you identify as weak points when solving problems
- Do the practice exam
  - As with the midterm, the practice exam gives you practice not only with topics but with the logistics of taking an exam on Gradescope

## Big O Quick Reference (see also http://bigocheatsheet.com/)

| What | Cost |
|---|---|
| • Hash table average case (good design) | O(1) |
| • Balanced trees<br>  • Heap, BST with balancing such as Red-Black<br>• Binary search on sorted array | O(logn) |
| • Linked list find<br>• Inserting into beginning of array/Vector<br>• Hash table worst case<br>• Unbalanced tree (e.g. unbalanced BST) worst case | O(n) |
| • Good sorting<br>  • Mergesort, Heapsort, Quicksort (expected) | O(nlogn) |
| • Bad sorting<br>  • Insertion, Bubble, Selection, Quicksort (worst case) | O($n^2$) |

## Quarter Wrap-Up

WHAT DID WE SET OUT TO DO IN THE BEGINNING?

WHERE ARE WE NOW?

## Goals for this Course

- **Learn how to model and solve complex problems with computers.**
- To that end:
  - Explore common abstractions for representing problems.
  - Harness recursion and understand how to think about problems recursively.
  - Bring added rigor to your understanding of algorithmic performance, so you can quantitatively compare approaches for solving problems.

## From here on out, there are no obvious answers to any problem worth your hourly rate. ☺

- Programming is all about exploring new ways to **model** and **solve** problems.

- There are **choices** and **tradeoffs** in how we model these and how we implement them!

- Skilled computer scientists recognize that any problem worth tackling has *many* possible models and *many* possible solutions, often none of which is clearly better than the others in all dimensions
  - Array or linked list? BST or hash table?
  - **Tradeoffs!**

## That’s a lot of material to cover in 10 weeks

You are part of a very challenging course,
in the best CS department *in the world*,
and you are so, so close to completing this course!

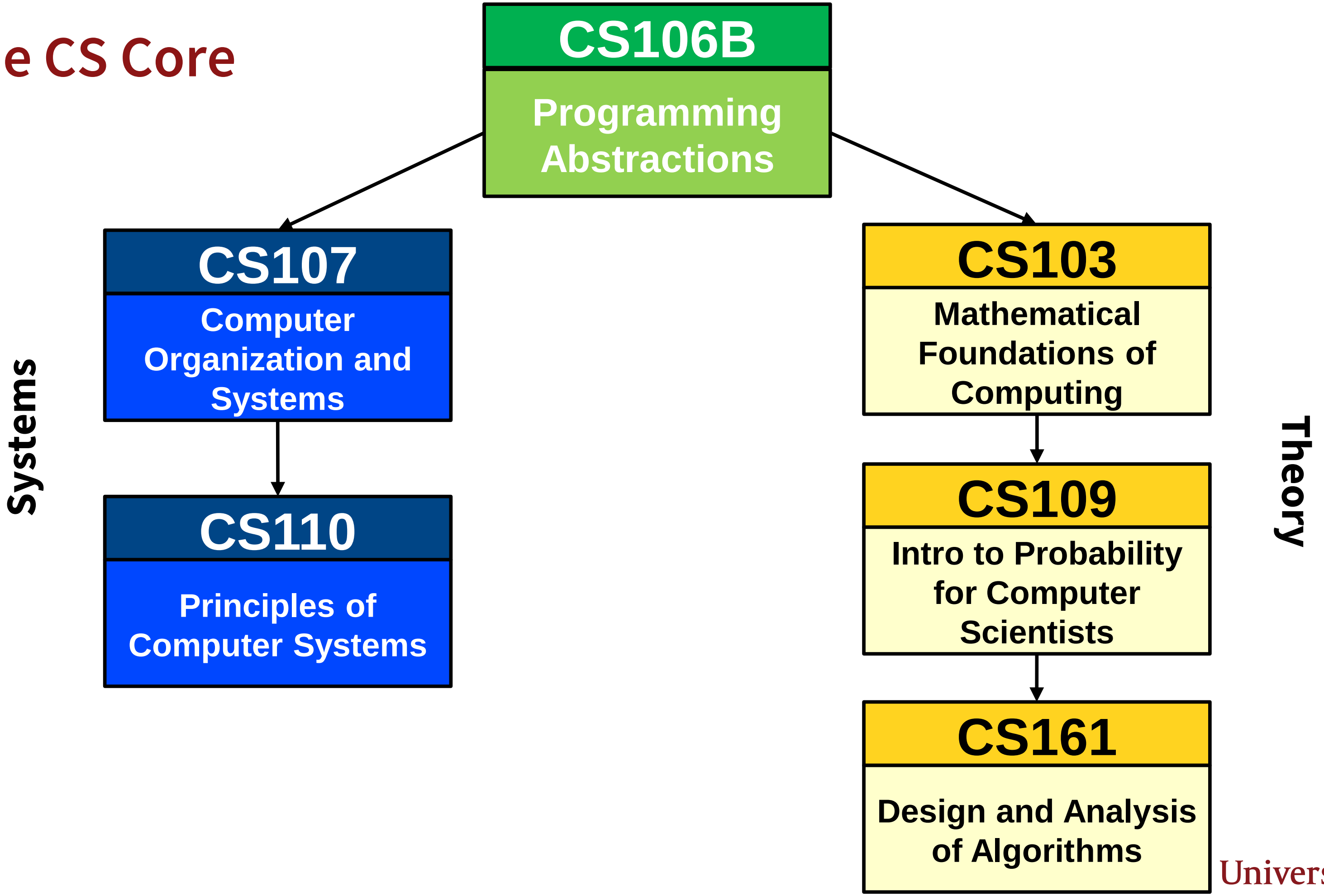
**Congratulations!!**
**You’ve almost made it through CS106B!**

- *So…what next?*

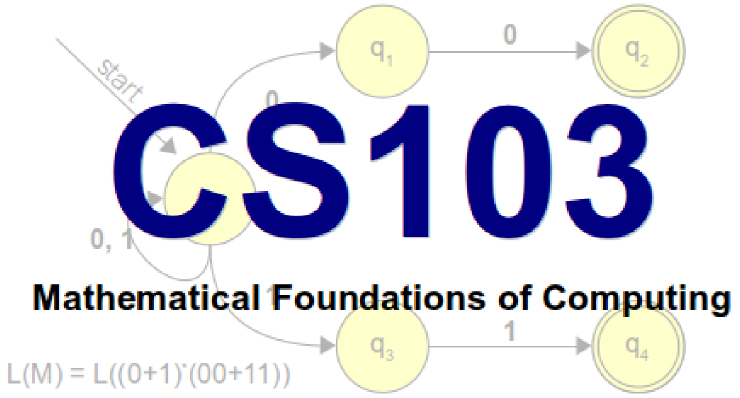
## What next?

CS OPTIONS AFTER CS106B

## The CS Core


CS106B
Programming Abstractions
CS107
Computer Organization and Systems
CS110
Principles of Computer Systems
CS103
Mathematical Foundations of Computing
CS109
Intro to Probability for Computer Scientists
CS161
Design and Analysis of Algorithms
Systems
Theory

University


CS103
Mathematical Foundations of Computing
start
0, 1
0
0
1
1
$q_1$
$q_2$
$q_3$
$q_4$
$L(M) = L((0+1)^*(00+11))$

## *Can computers solve all mathematical problems?*

Spoiler: no!

## *Why are some problems harder than others?*

We can do find in an unsorted array in O(N), and we can sort an unsorted array in O(NlogN). Is sorting just inherently a harder problem, or are there better O(N) sorting algorithms yet to be discovered?

## *How can we be certain about this?*

Stanford University

## CS107
## Computer Organization and Systems

**How do we encode text, numbers, programs, etc. using just 0s and 1s?**

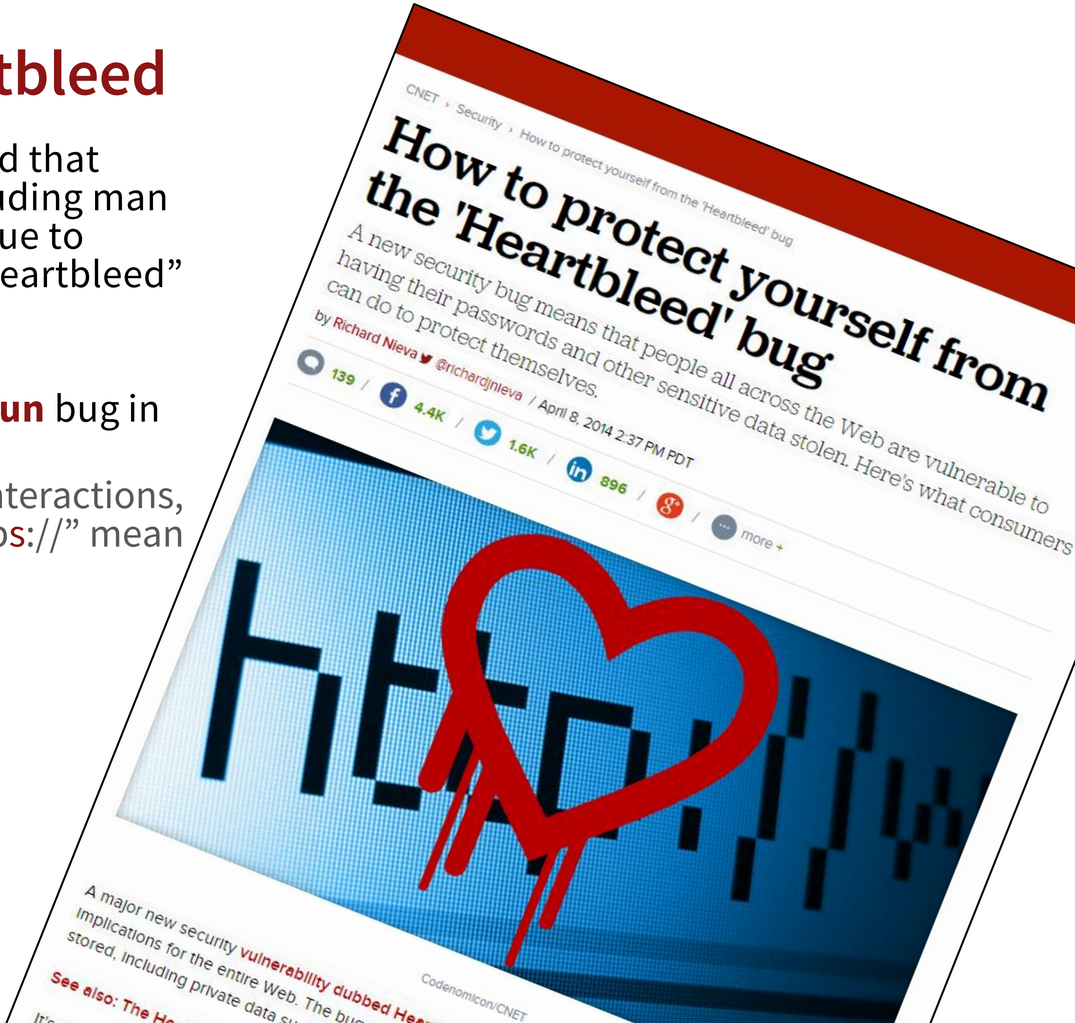
**Where does memory come from? How is it managed?**

**How do compilers, debuggers, etc. work?**

## CS107 in the news: Heartbleed

- In April 2014, security experts warned that users of thousands of websites, including man needed to change their passwords due to potential exposure caused by the "Heartbleed" vulnerability

- Heartbleed exploited a **buffer overrun** bug in OpenSSL
  - SSL is the layer that secures web interactions, i.e., it's what makes the "s" in "https://" mean something


CNET › Security › How to protect yourself from the 'Heartbleed' bug

How to protect yourself from the 'Heartbleed' bug

A new security bug means that people all across the Web are vulnerable to having their passwords and other sensitive data stolen. Here's what consumers can do to protect themselves.

by Richard Nieva @richardjnieva / April 8, 2014 2:37 PM PDT

139 / 4.4K / 1.6K / 896 / more +

Codenomicon/CNET

A major new security vulnerability dubbed Hea
implications for the entire Web. The bu
stored, including private data s

See also: The He

It's

## CS107 in the news: Heartbleed

- The protocol allows you to send “heartbeat” messages, which basically say:
  - *`Are you still there? If you are, repeat this message back to me: "hello" [0x0005 bytes].`*
  - Each char is one byte, so 5 letters

- Unfortunately, the software also let you send messages like this:
  - *`Are you still there? If you are, repeat this message back to me: "hello" [0xFFFF bytes].`*
  - That’s 65535 bytes—much more than the length of `"hello"`!
  - So the software would continue for-looping past the end of the `"hello"` array, sending information back
  - Which causes an error, right? **RIGHT??** Turns out, no.

CNET › Security › How to protect yourself from the 'Heartbleed' bug

**How to protect yourself from the 'Heartbleed' bug**

A new security bug means that people all across the Web are vulnerable to having their passwords and other sensitive data stolen. Here's what consumers can do to protect themselves.

by Richard Nieva @richardjnieva / April 8, 2014 2:37 PM PDT

139 / 4.4K / 1.6K / 896 / / more +

Codenomicon/CNET

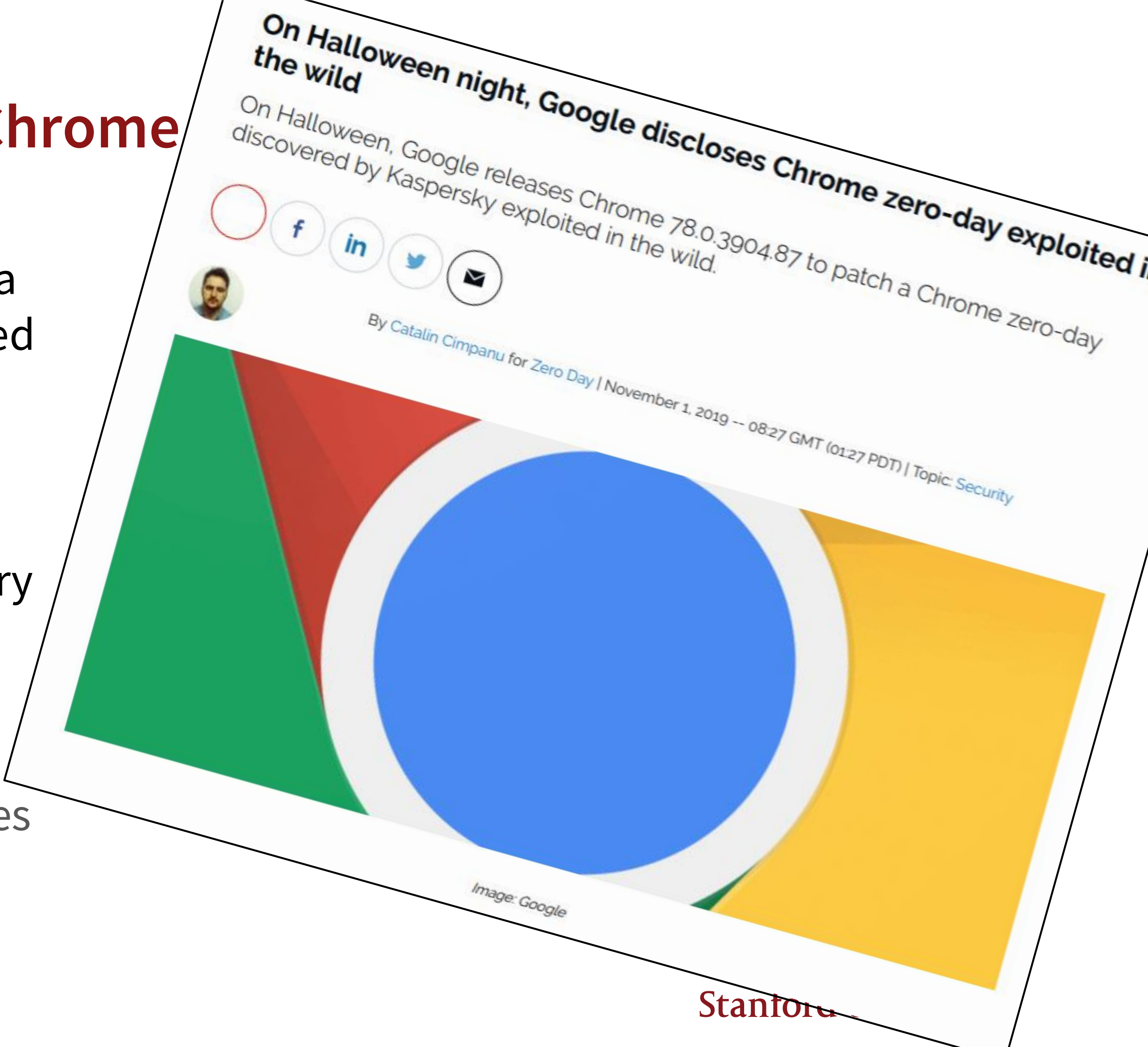
A major new security vulnerability dubbed Hea
implications for the entire Web. The bu
stored, including private data su

See also: The He

It'

## CS107 in the news: Chrome

- On Oct 31, 2019, Google disclosed that there was a bug in Chrome that caused a security breach
- The bug was that the program accesses memory after it has already been freed/deleted
  - Usually works, but incorrect and sometimes causes the bug



**On Halloween night, Google discloses Chrome zero-day exploited i the wild**

On Halloween, Google releases Chrome 78.0.3904.87 to patch a Chrome zero-day discovered by Kaspersky exploited in the wild.

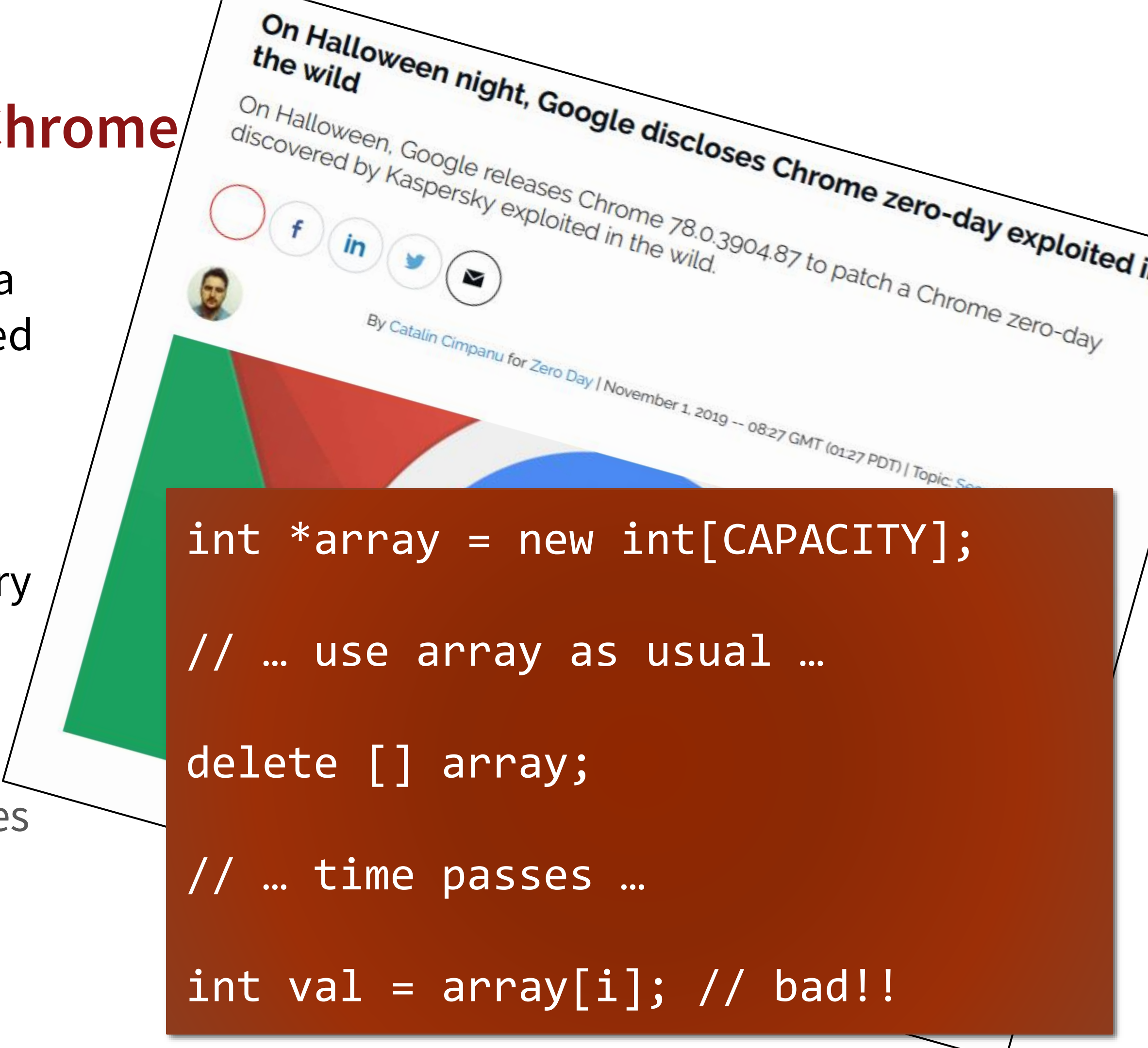
By Catalin Cimpanu for Zero Day | November 1, 2019 -- 08:27 GMT (01:27 PDT) | Topic: Security

Image: Google

Stanford

## CS107 in the news: Chrome

- On Oct 31, 2019, Google disclosed that there was a bug in Chrome that caused a security breach
- The bug was that the program accesses memory after it has already been freed/deleted
  - Usually works, but incorrect and sometimes causes the bug



On Halloween night, Google discloses Chrome zero-day exploited i
the wild

On Halloween, Google releases Chrome 78.0.3904.87 to patch a Chrome zero-day discovered by Kaspersky exploited in the wild.

By Catalin Cimpanu for Zero Day | November 1, 2019 -- 08:27 GMT (01:27 PDT) | Topic: Se

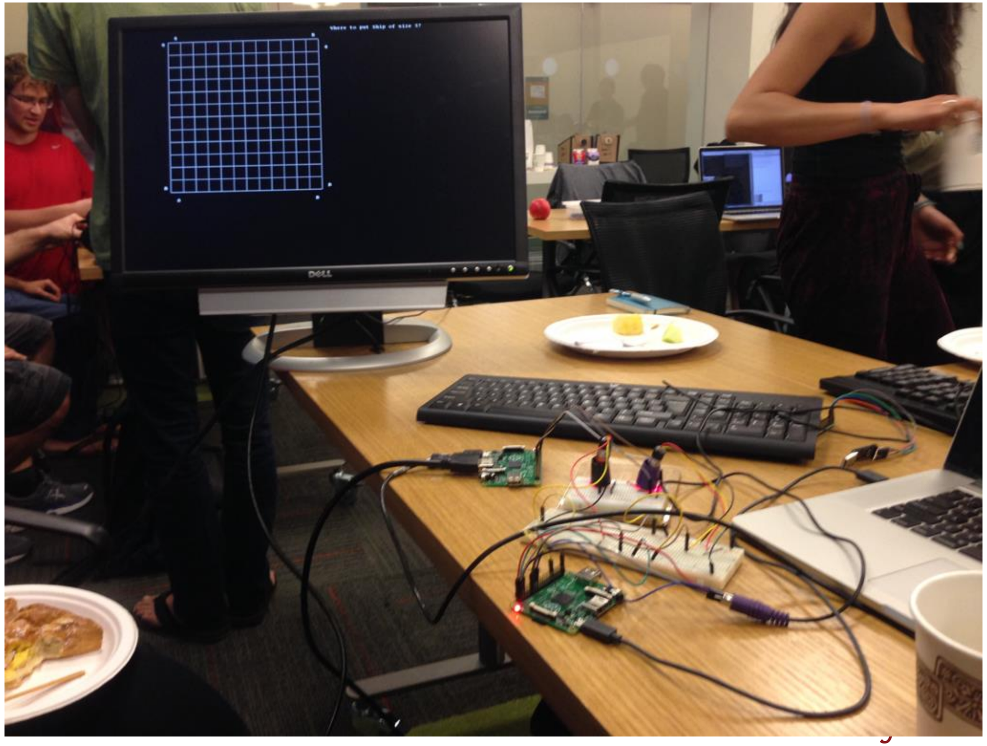
```
int *array = new int[CAPACITY];

// … use array as usual …

delete [] array;

// … time passes …

int val = array[i]; // bad!!
```

## What CS107 Isn't

- CS107-style systems programming is ***one*** kind of programming.
  - One that I really love! But one of many, and it’s a matter of personal taste.
- CS107 is ***not*** a litmus test for whether you can be a computer scientist.
  - You can be a *great* computer scientist without particularly enjoying low-level systems programming.
- CS107 is ***not*** indicative of what programming is “really like.”
  - CS107 does a lot of low-level programming. You don't have to do low-level programming to be a good computer scientist.

## CS107E
## Computer Organization and Systems—*Embedded*

- **Counts for prerequisites etc. the same as original CS107**, but covers the topics with a twist: embedded work on Raspberry Pi



## Other CS Courses

## CS106L
## Learning the Standard Template Library (STL)

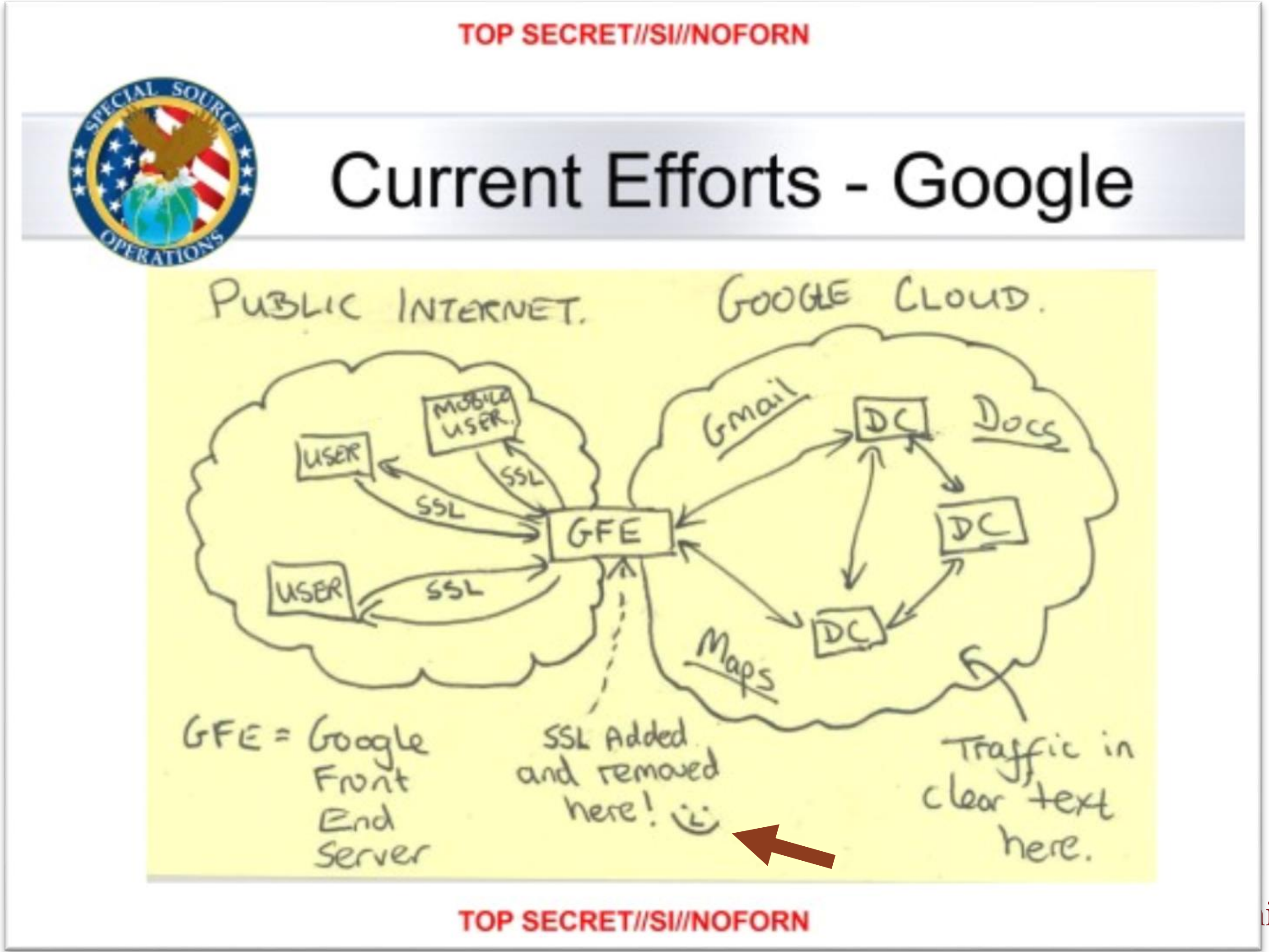
- In CS106B, we learn the Stanford Library containers
- ***Now learn the industrial-strength ones!***

## CS182
## Computers, Ethics, and Public Policy

- Did you love the thought-provoking issues raised in Katie Creel's lecture in this class?
- Want to dive deeper into that topic and more, and be prepared to participate in urgent, ongoing conversations in media, government, non-profits, and the tech industry?

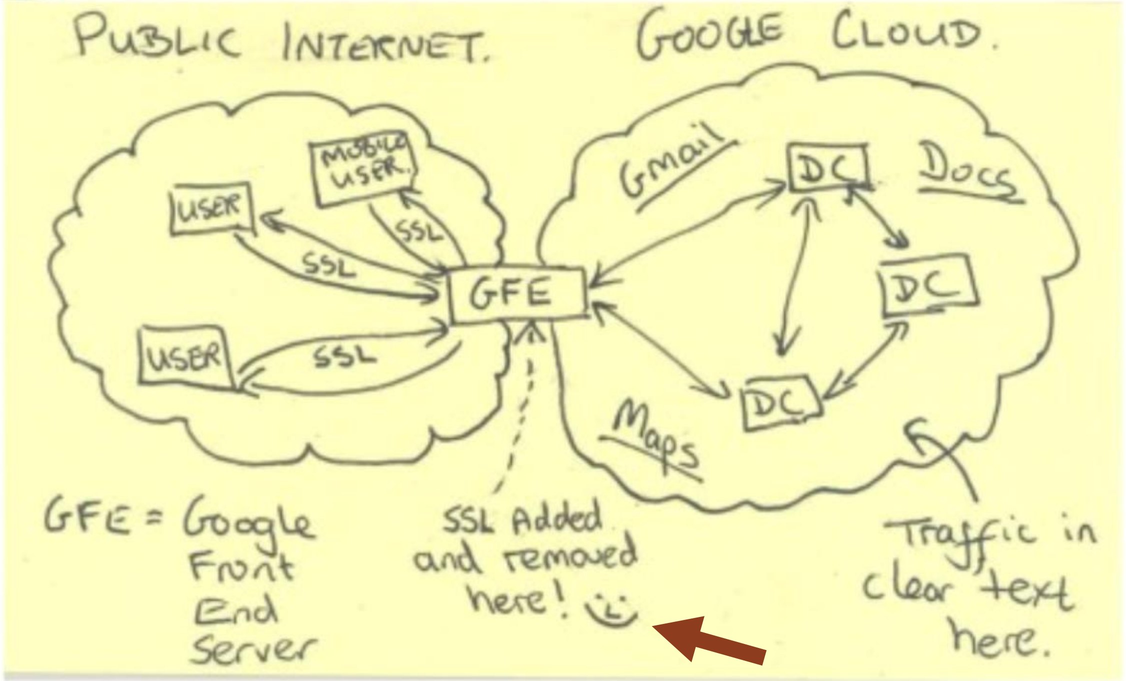
***We have the power to control and create technology, but how should we use it? Who should get a say in how it's used?***

Stanford University



TOP SECRET//SI//NOFORN

# Current Efforts - Google


PUBLIC INTERNET.
GOOGLE CLOUD.
MOBILE USER.
USER
USER
SSL
SSL
SSL
GFE
Gmail
DC
Docs
DC
DC
Maps
GFE = Google Front End Server
SSL Added and removed here! :-)
Traffic in clear text here.

TOP SECRET//SI//NOFORN

## CS108
## Object-Oriented Systems Design

- ***How do you build large software systems in a team?***
- Introduction to things you need to know for work in the “real world”:
  - Unit-testing frameworks
  - Object-oriented design
  - Multithreaded applications
  - Databases and web applications
  - Source control

## Options besides CS Major

**CS Minor: only 5 more classes!**

- 103, 107, 109, two your choice—fun!

**CS Coterminal MS degree**

- Earn an MS in CS while you are here earning your BS
- Possible for CS majors **and** other majors
  - ex: Psych major, CS co-term

See you Wednesday!