The Future of Computing and the Future of 15-213

15-213: Introduction to Computer Systems
Last Lecture, Aug. 1, 2022

Logistics – Last Two Weeks of Class

- **This is the last official lecture**
  - Remaining class sessions are open TA+faculty office hours
- **Proxy lab checkpoint is due August 4**
- **Proxy lab final is due August 11**
  - NO GRACE DAYS
  - Any extension is technically an incomplete
- **Exam is August 10, here, 12:30-3:30pm**
  - Disability accommodations? Go to ODR Testing Center instead
- **Prof. Railing is on vacation**
  - Prof. Weinberg's email is zackw@cmu.edu
  - Prof. Weinberg's office is GHC 4124
  - I'll be there after class till 4pm, every day this week and next, except:
    - Thursday Aug. 3
    - Thursday Aug. 10
    - Friday Aug. 11

50 Years of "Moore's Law"

Transistor count
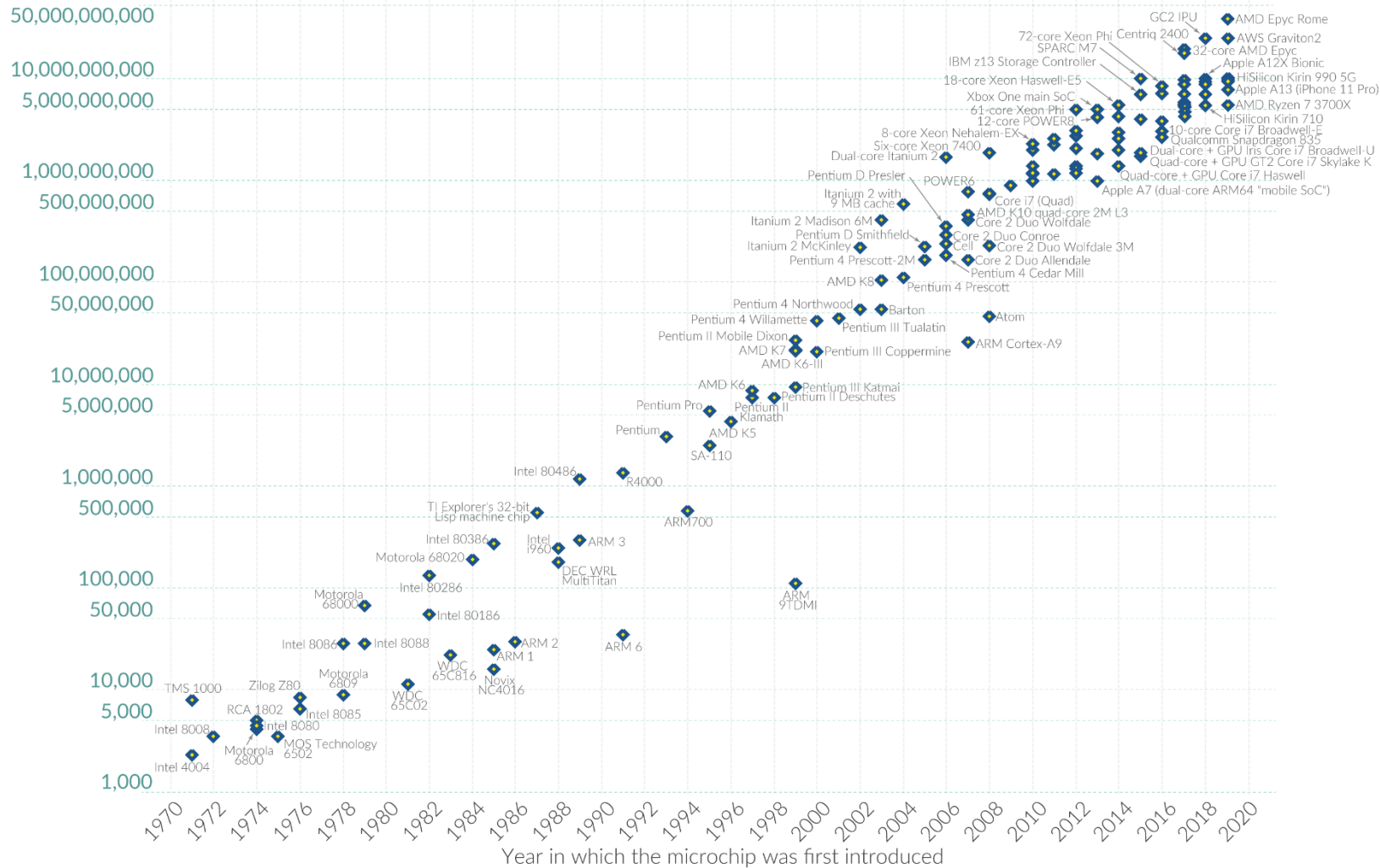
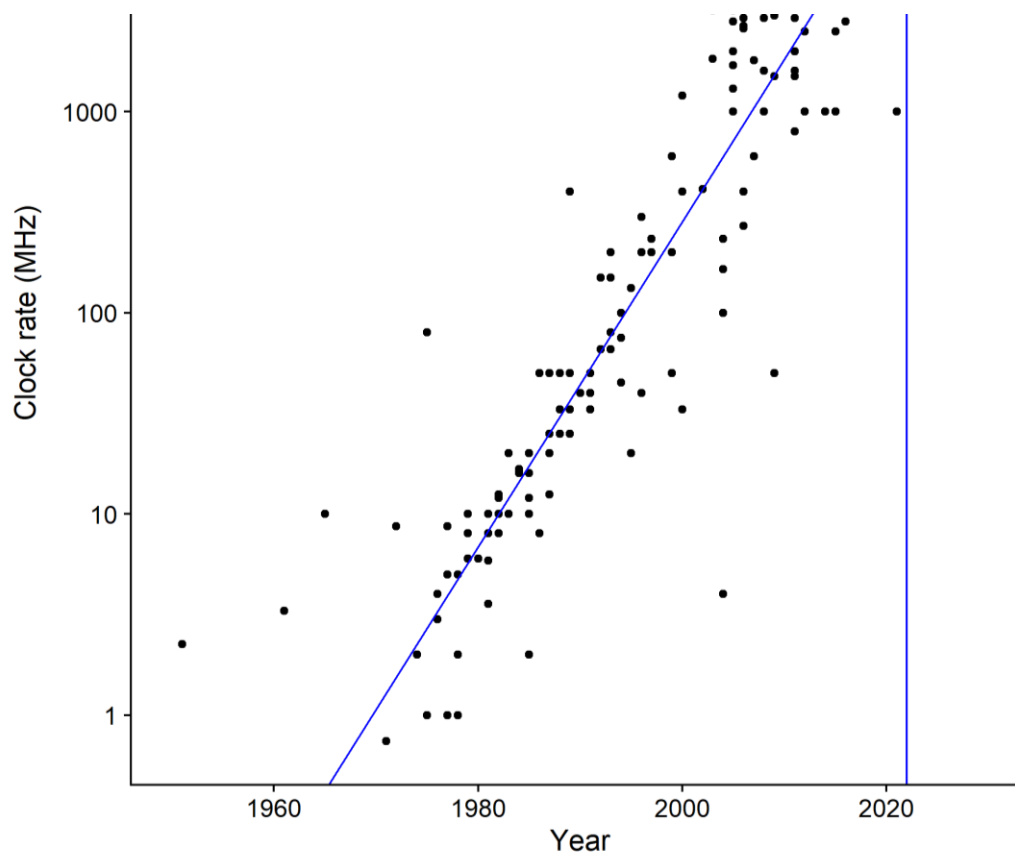# What Moore's Law Has Meant

■ **1976 Cray 1**

- 250 M Ops/second
- ~170,000 chips
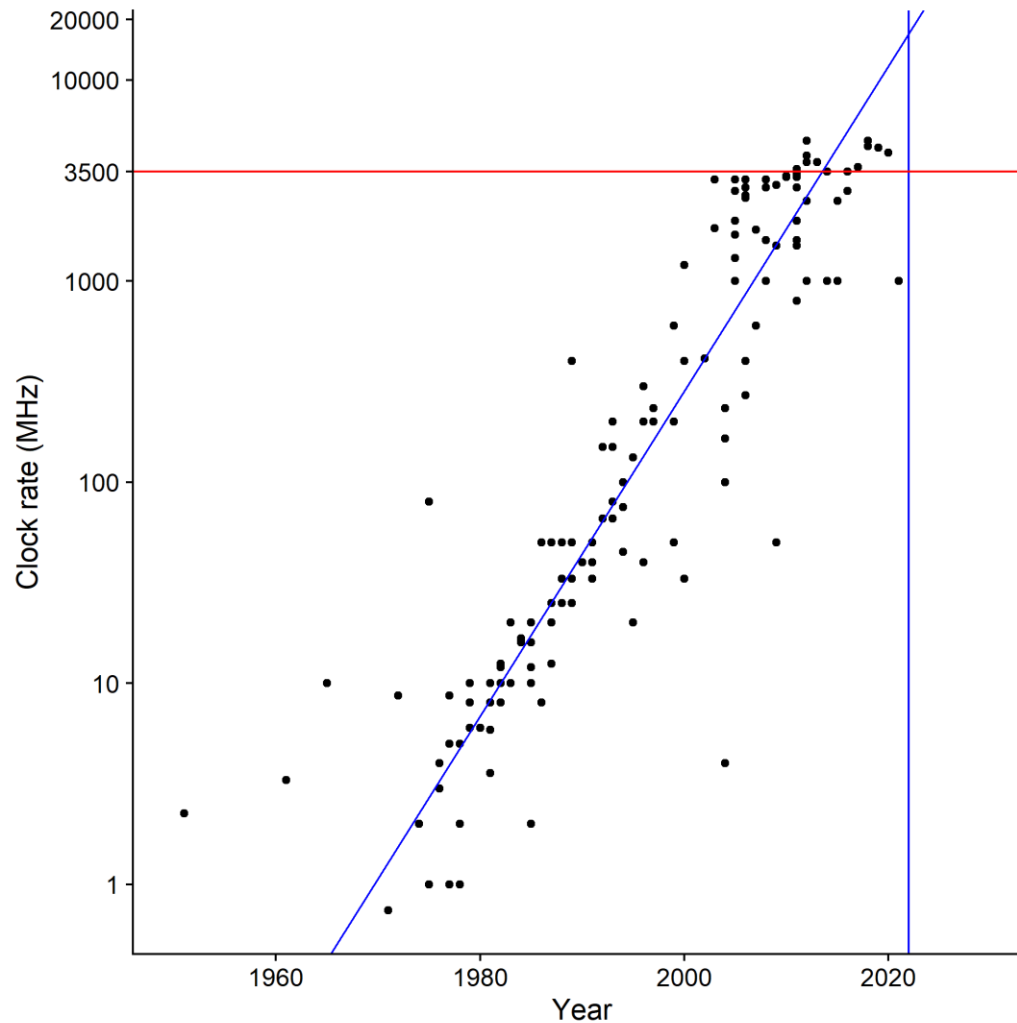- 0.5B transistors
- 5,000 kg, 115 KW
- $9M
- 80 manufactured

■ **2014 iPhone 6**

- > 4 B Ops/second
- ~10 chips
- > 3B transistors
- 120 g, < 5 W
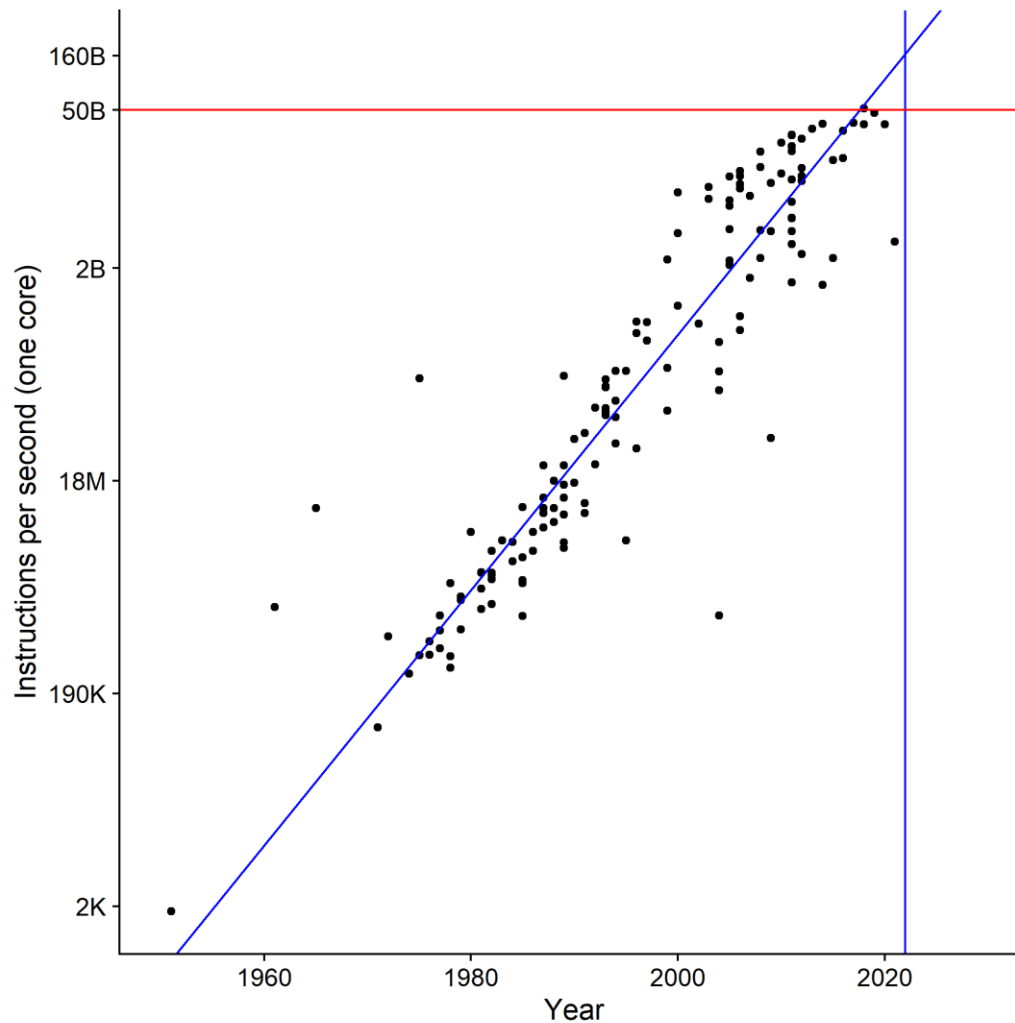- $649
- 10 million sold in first 3 days
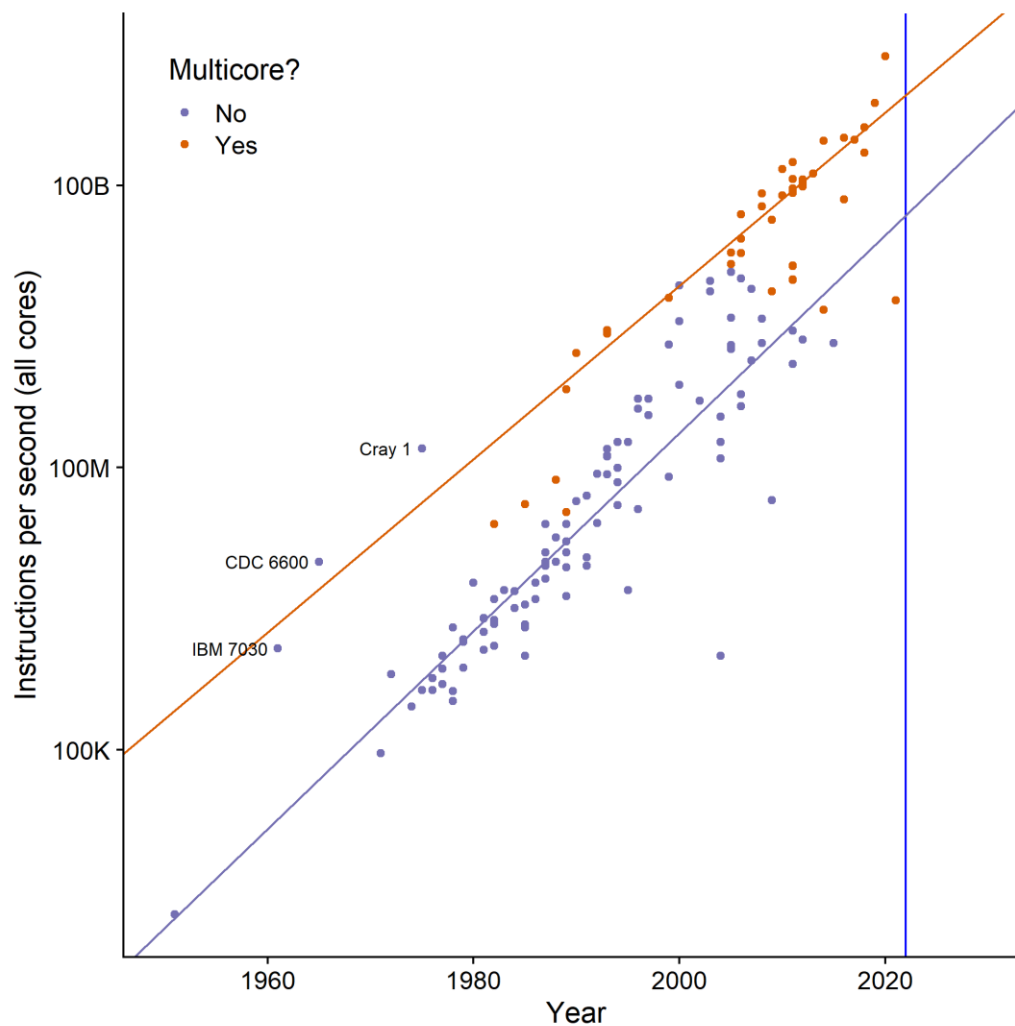
Exponential trends can't continue forever

Exponential trends can't continue forever

Clock rate → single thread performance

If all else fails, add parallelism…

Parallel Computation and 213

- **Last week we had *classes* on parallel programming**
- **In shell lab you have to think about concurrent processes**
- **In proxy lab you will use threads to serve many simultaneous clients**
  - But the concurrency isn't really the *point* of either lab
- **We're working on a new lab where concurrency is the most important thing**
  - Also covers file system semantics and the craft of modifying existing code
  - Not done, but we'd like to show it to you anyway…

The Shark File System Lab

- **Requin wrote a file system to store his files**

- **He needs your help to finish it**
  - Several important functions are not implemented
  - Only one program can access the file system at a time
  - There's some other problems (we'll let you discover them ;-)

- **Handed out on Autolab / Github Classroom as usual**
  - https://autolab.andrew.cmu.edu/courses/15213-m23/assessments/sfslab
  - 213 students only
  - Released Wednesday evening

- **NOT FOR CREDIT, just for fun**
  - But it may help with the final
  - You might want to finish proxy lab first
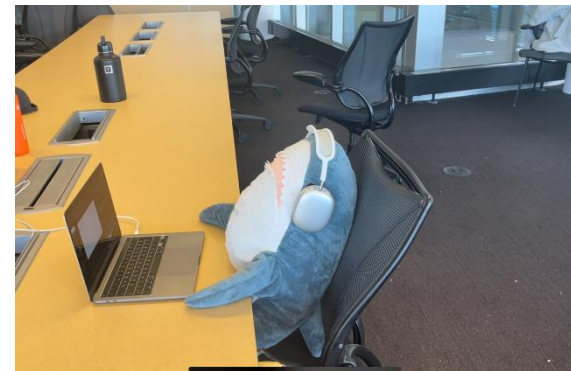
Requin hard at work

SFS Lab Overview

sfslab-handout $ ls

`Makefile`	*The usual helper files…*
`README`	
`check-format`	
`helper.mk`	
`lua/`	*Interpreter for the Lua scripting language (used in traces)*
`sfs-disk.c`	***This is the file you'll need to modify***
`sfs-disk.h`	*Interface between sfs-disk.c and sfs-tester.c*
`sfs-driver.py`	*Runs sets of traces*
`sfs-tester.c`	*Runs one trace; can also be used interactively*
`traces/`	*Our tests for your code (more to come…)*
`YOUR-FEEDBACK.md`	***Write your thoughts on this lab here and 'make submit' it***

SFS Lab Traces

- **Like proxy lab, traces are in four categories**

- **S traces test *single-threaded correctness***
  - You can pass these traces by implementing the functions `sfs_getpos`, `sfs_seek`, and `sfs_rename`

- **C traces test *concurrent correctness***
  - To pass these traces you will need to revise sfs-disk.c and make it thread-safe

- **P traces test *concurrent performance***
  - There aren't any P traces yet

- **X traces test *optional additional features***
  - For people who want to go beyond our basic expectations
  - More to do with filesystem capabilities than with concurrency

SFS Lab Logistics

- **We're still actively designing this lab**
- **We will be updating the handout over the next two weeks**
  - Use "make update-handout" to pull updates
- **We want your suggestions and commentary**
  - Write whatever you want us to see in the "YOUR-FEEDBACK.md" file, "make submit" will send it to us
  - Write your own trace files! "make submit" will also send those in
- **Feel free to change ANYTHING**
  - Except for the "lua" subdirectory, which we didn't write
  - (You're encouraged to *read* the code in the "lua" subdirectory, especially if you are curious about how interpreters work)

SFS Lab Resources

- **Guides for concurrent programming with C and pthreads:**
  - https://randu.org/tutorials/threads/
  - https://web.archive.org/web/20180512055837/https://www.cs.nmsu.edu/~jcook/Tools/pthreads/pthreads.html
  - https://www.mit.edu/people/proven/IAP_2000/index.html
  - I don't love any of these. If you find a better one please tell us about it.
- **Documentation for the Lua scripting language:**
  - https://www.lua.org/docs.html
  - https://lualanes.github.io/lanes/
- **The only documentation for SFS itself is the writeup and comments in sfs-disk.c, sfs-disk.h, and sfs-tester.c.**
  - We regret the inconvenience.
  - Do not hesitate to ask questions (on Piazza, in class, in TA office hours, show up to Prof. Weinberg's office any time the door is open)

Can't outrun Amdahl

- **Moore:**
  - Most CPU functions got faster simultaneously
  - Memory density scaled too!
  - I/O (& mem latency) was the primary bottleneck to work around
- **Multicore:**
  - Only some algorithms can be parallelized
- **GPUs / SIMD**
  - Only some algorithms can be vectorized
- **Special purpose instructions, custom ASICs**
  - Only good for that one thing

$$S = \frac{1}{(1 - p) + \frac{p}{s}}$$

$$\lim_{b \to 0} a + b = a$$

This machine costs 150 million dollars

Exponential capital cost → exponentially fewer manufacturers

130nm	90nm	65nm	45/40nm	32/28nm	22/20nm
Altis Semiconductor					
Dongbu HiTek	Dongbu HiTek				
Grace Semiconductor	Grace Semiconductor				
SMIC	SMIC				
UMC	UMC				
TSMC	TSMC	SMIC			
Globalfoundries	Globalfoundries	UMC			
Seiko Epson	Seiko Epson	TSMC			
Freescale	Freescale	Globalfoundries	SMIC		
Infineon	Infineon	Infineon	UMC		
Sony	Sony	Sony	TSMC		
Texas Instruments	Texas Instruments	Texas Instruments	Globalfoundries		
Renesas (NEC)	Renesas	Renesas	Renesas		
IBM	IBM	IBM	IBM	UMC	
Fujitsu	Fujitsu	Fujitsu	Fujitsu	TSMC	
Toshiba	Toshiba	Toshiba	Toshiba	Globalfoundries	TSMC
STMicroelectronics	STMicroelectronics	STMicroelectronics	STMicroelectronics	STMicroelectronics	Globalfoundries
Intel	Intel	Intel	Intel	Intel	Intel
Samsung	Samsung	Samsung	Samsung	Samsung	Samsung

17

The future of computing isn't (only) about speed anymore.

:(

Your PC ran into a problem that it couldn't handle, and now it needs to restart.

You can search for the error online: HAL_INITIALIZATION_FAILED

```
*** STOP: 0x00000050 (0xFCCBFFFF,0x00000000,0x00000000,0x00000000)
PAGE_FAULT_IN_NONPAGED_AREA

p5-0000  irql:2   SYSVER 0xf0000421

Dll Base DateStmp - Name                    Dll Base DateStmp - Name
80100000 2fc653bc - ntoskrnl.exe            80400000 2fb24f68 - hal.dll
80010000 2fc28575 - atapi.sys               80013000 2faae8ca - SCSIPORT.SYS
8001b000 2faae85d - BusLogic.sys            80001000 2faae8c5 - Scsidisk.sys
80f00000 2fc4f4b2 - Ntfs.sys                fcc00000 2faae8af - Floppy.SYS
fcc10000 2fb16eef - Scsicdrm.SYS            fcc20000 2faae8a2 - Cdaudio.SYS
fcc30000 2faae8ff - Fs_Rec.SYS              fcc40000 2faae8b7 - Null.SYS
fcc50000 2faae8a1 - Beep.SYS                fcc60000 2faae8de - sndblst.SYS
fcc70000 2faae8b2 - i8042prt.SYS            fcc80000 2faae8b5 - Mouclass.SYS
fcc90000 2faae8b4 - Kbdclass.SYS            fccb0000 2faae88d - VIDEOPRT.SYS
fcca0000 54a49c11 - vbemp.sys               fccc0000 2faae892 - vga.sys
fccd0000 2faae8fd - Msfs.SYS                fcce0000 2faae8ec - Npfs.SYS

Address   dword dump    Build [1057]                               - Name
fcc2caf4 8013b26b 8013b26b 00000000 fccbffff 00000000 01450144 - ntoskrnl.exe
fcc2cbc8 80102f48 80102f48 fde54c40 fde9fee8 804024d8 fde54c40 - ntoskrnl.exe
fcc2cbd4 804024d8 804024d8 fde54c40 8015ccca fdea86d0 fcc2cd4c - hal.dll
fcc2cbdc 8015ccca 8015ccca fdea86d0 fcc2cd4c fde54c28 00000018 - ntoskrnl.exe
fcc2cc14 8015c542 8015c542 e118bf58 00000019 fcc2ce1c fcc2cccc - ntoskrnl.exe
fcc2cc4c 8019204d 8019204d fcc2ce18 00120089 00120089 fcc2ce28 - ntoskrnl.exe
fcc2cc90 80134f30 80134f30 80100f60 ffffffff fcc2cd0c 80179b00 - ntoskrnl.exe
fcc2cc94 80100f60 80100f60 ffffffff fcc2cd0c 80179b00 fde54c40 - ntoskrnl.exe
fcc2cca0 80179b00 80179b00 fde54c40 00000000 fcc2cd4c c2ce0801 - ntoskrnl.exe
fcc2cd10 80175cf6 80175cf6 00000000 fcc2cdc4 00000040 00000000 - ntoskrnl.exe
fcc2cd64 80100080 80100080 00000000 00000000 e1181030 fde4b020 - ntoskrnl.exe
fcc2cde8 80151e33 80151e33 0021f378 00000000 c2ce0801 00000000 - ntoskrnl.exe
fcc2cdfc 80100080 80100080 fcc2ce14 fcc2ce68 fcc2cf04 0021f36c - ntoskrnl.exe
fcc2ce10 801560fe 801560fe 00480008 fde48a88 00000000 801246cb - ntoskrnl.exe
fcc2ce20 801246cb 801246cb 00000000 00000000 00000000 00000000 - ntoskrnl.exe
fcc2ce7c 80177c08 80177c08 ffffffff 80134f30 80100c60 ffffffff - ntoskrnl.exe
fcc2ce84 80134f30 80134f30 80100c60 ffffffff fcc2ced0 8015612d - ntoskrnl.exe
fcc2ce88 80100c60 80100c60 ffffffff fcc2ced0 8015612d 0021f3d4 - ntoskrnl.exe
fcc2ce94 8015612d 8015612d 0021f3d4 80100080 0021f378 0021f3b8 - ntoskrnl.exe
fcc2ce9c 80100080 80100080 0021f378 0021f3b8 00000000 00000080 - ntoskrnl.exe
fcc2ced4 80137fb5 80137fb5 0021f3d4 80100080 0021f378 0021f3b8 - ntoskrnl.exe
fcc2cedc 80100080 80100080 0021f378 0021f3b8 00000000 00000080 - ntoskrnl.exe

Riavviare e impostare le opzioni di ripristino nel pannello di controllo del
sistema o l'opzione di avvio del sistema /CRASHDEBUG. Se il messaggio riappare
rivolgersi all'amministratore del sistema o al gruppo di supporto tecnico.
```
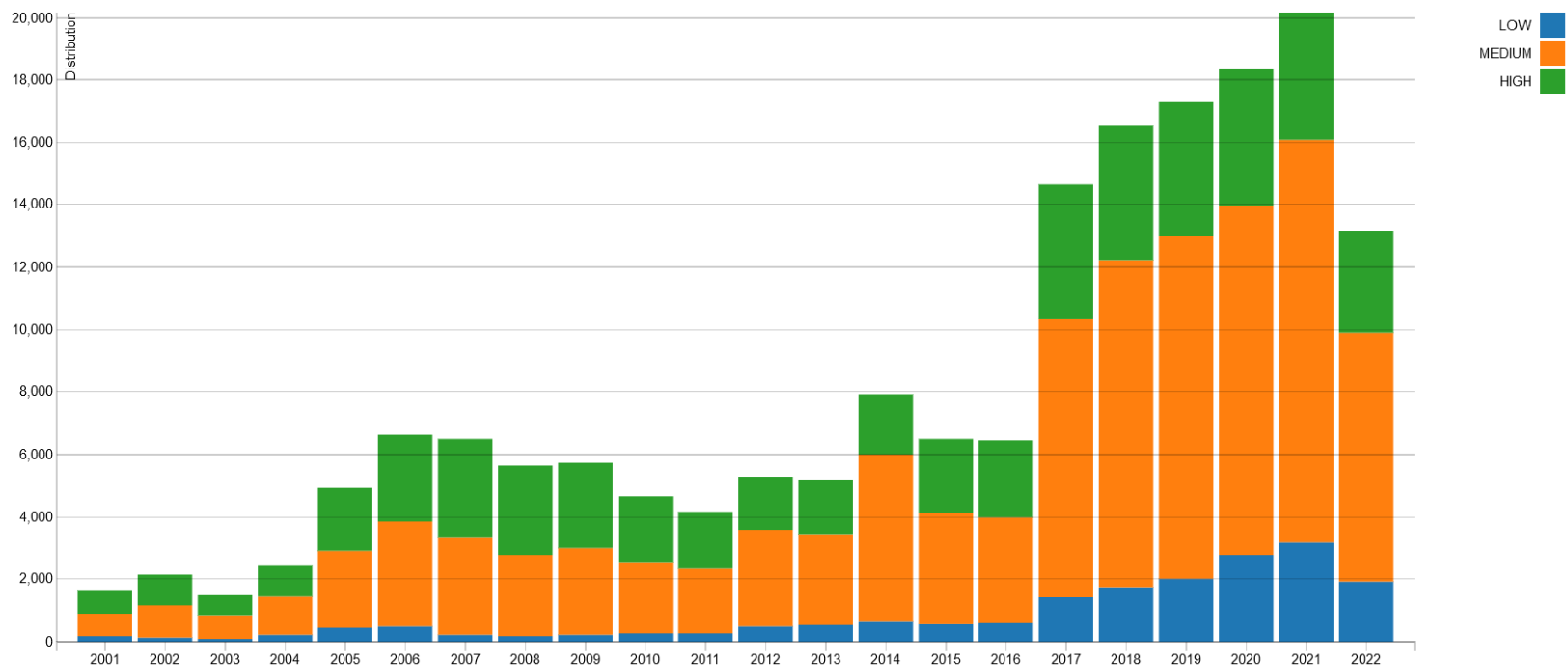
Bugs reported to the National Vulnerability Database (USA), 2001–present (only security-critical bugs are counted)

Actual bug in attack lab phase 3

Buggy

```
void touch3(char *sval)
{
  if (hexmatch(cookie, sval)) {
    printf(
      "Touch3!: You called touch3(\"%s\")\n",
      sval);
    validate(3);
  } else {
    printf(
      "Misfire: You called touch3(\"%s\")\n",
      sval);
    fail(3);
  }
  exit(0);
}
```

Fixed

```
void touch3(char *sval)
{
    if (hexmatch(cookie, sval)) {
        report_touch3("Touch3!", sval);
        validate(3);
    } else {
        report_touch3("Misfire", sval);
        fail(3);
    }
    exit(0);
}

void report_touch3(const char *tag, const char *sval) {
    const char *p;
    printf("%s: You called touch3(\"", tag);
    for (p = sval; *p; p++) {
        unsigned char c = (unsigned char)*p;
        if (c < ' ' || c > '~')
            printf("\\x%02x", c);
        } else if (c == '\\') {
            putchar('\\');
            putchar('\\');
        } else if (c == '"') {
```

Growth of Average Web Page Size and Number of Objects - Jan 1995-July 2014
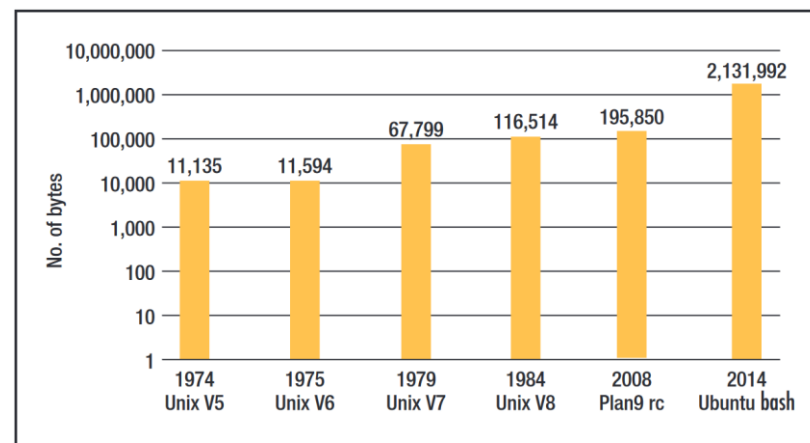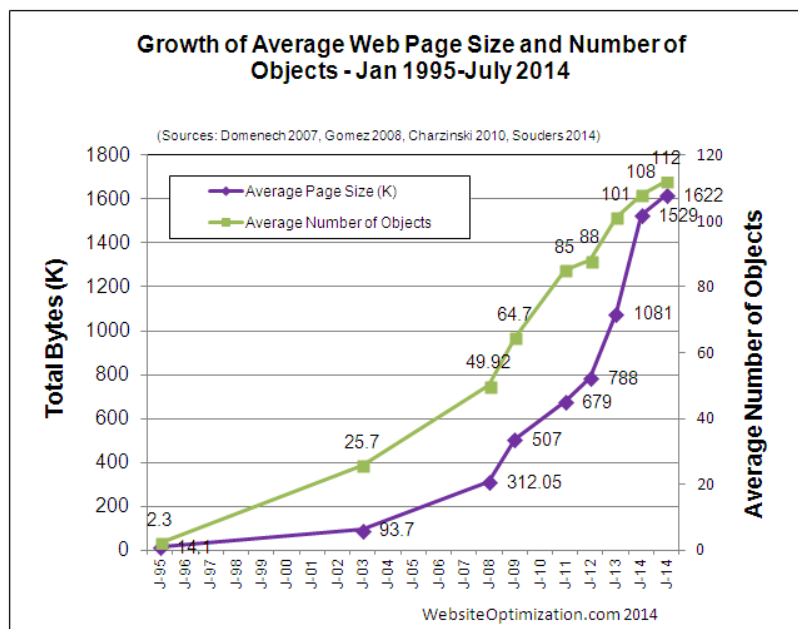
FIGURE 2. The source code size over time for the default command shell on Unix and Unix-like systems. From approximately 11 Kbytes in fifth-edition Unix in 1974 to 2.1 Mbytes for the bash shell 40 years later is an increase of 191 times.

The local newspaper website...

1166 HTTP requests
47 MB of data transferred
Still loading stuff in the background 1 minute after this screen stabilized

… without the ads

87 HTTP requests
7.4 MB of data transferred
No network activity after 1.5 seconds

Conflicting incentives

- **I want to read the news**
  - I don't want ads distracting me from the news
  - I don't want to wait for irrelevant pictures to load
- **I want to discuss the news with my friends**
  - I want to be able to share individual articles with them
  - They should be able to read the whole thing without subscribing
- **I want the subscription fee to be small**

- **The newspaper needs money to operate**
  - Salaries for reporters, editors, photographers
  - Website hosting fees
  - Print and distribute paper edition
- **Subscription fees have never been sufficient**
  - Advertising makes up the difference
  - Advertising means newspaper answers to advertisers

"*algorithm*???
that thing where
social media sites get money from advertisers
the more shit you click on
so they shove content at you
that is theoretically to your interest
but actually is content that will keep you clicking,
including content that enrages you,
because they don't care about giving you shit you like
just about keeping you engaged as long as possible,
and oh also did we mention
this ties into the modern lack of data privacy,
the algorithm is watching everything you interact with
to profile you better for advertising to you,
it's spying on you across platforms whenever possible,"
— @jmtorres on Tumblr

Not all problems have technical solutions

Evi Nemeth's extended OSI protocol stack, quoted in https://www.linuxjournal.com/article/10726

COMPLEXITY   OR SIMPLICITY ?

Image from ch. 11 of https://www.forth.com/starting-forth/
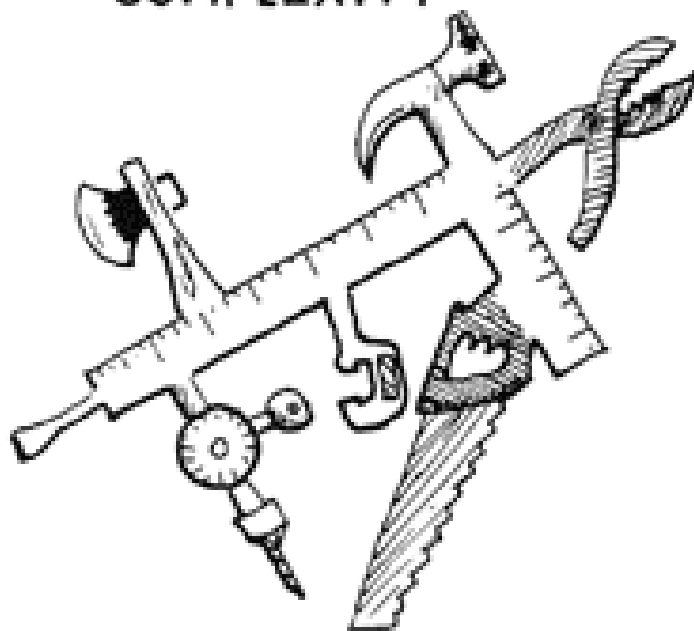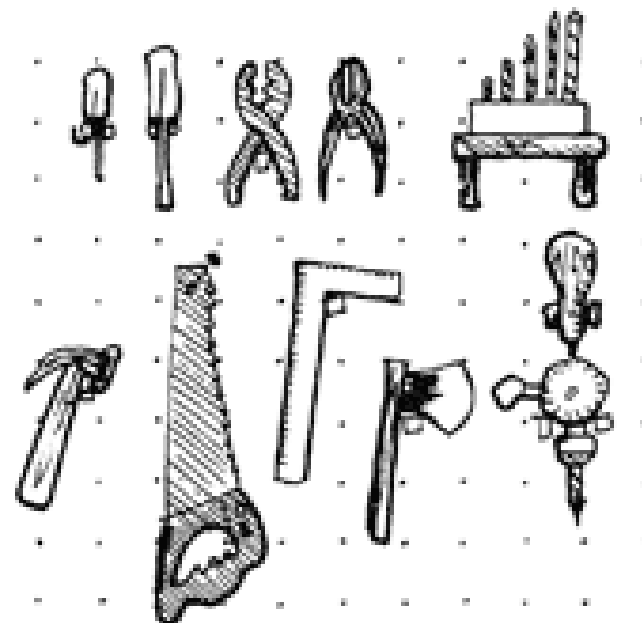
Customizability

This is what my text editor looks like.

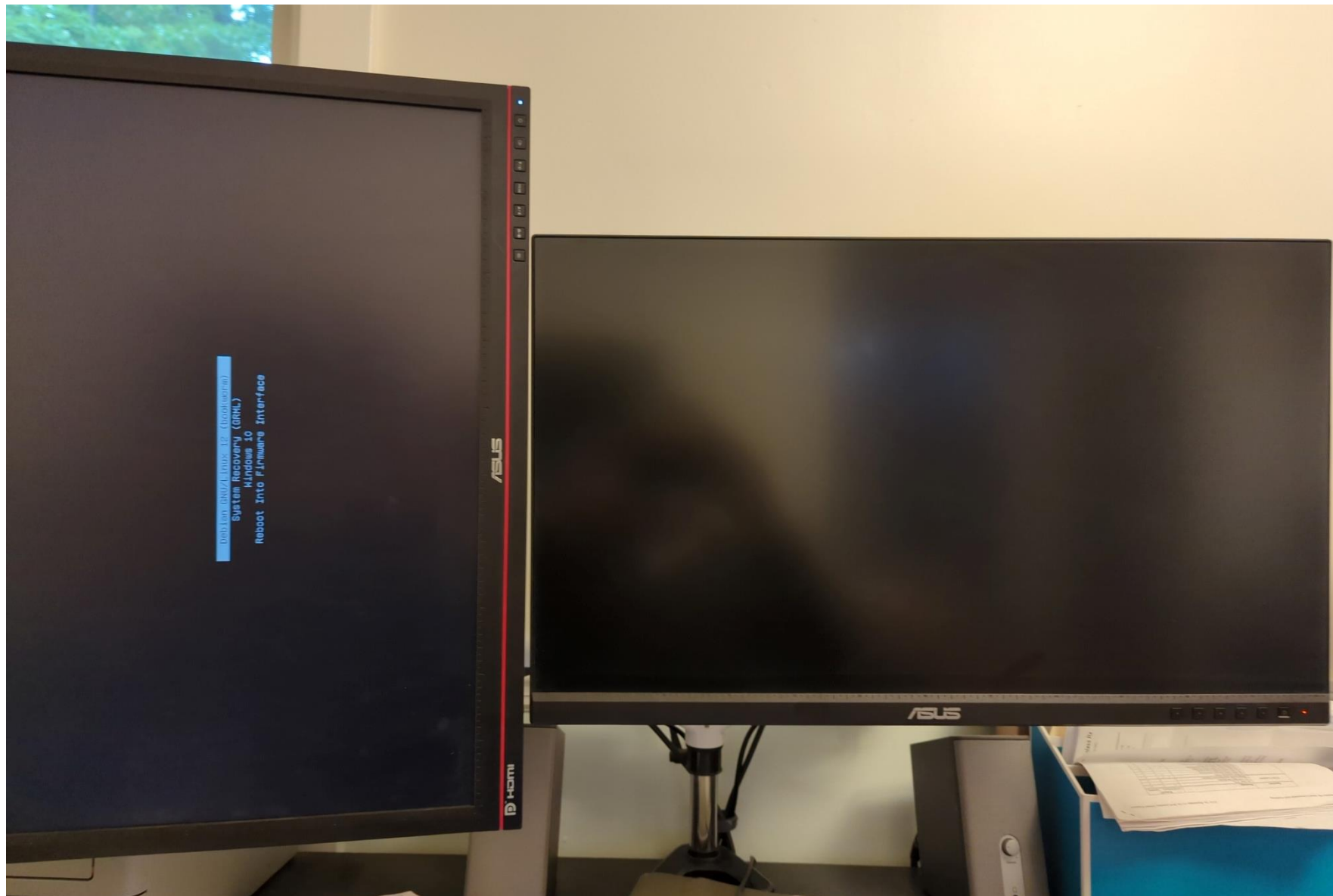**I write *everything* (except PowerPoint decks ;-) in this interface.**

Customizability

- **This is what the same editor looks like if I start it up without any customization.**

- **I've written hundreds of lines of Lisp to get this thing exactly the way I want it.**

- **Most of that is about behavior, not appearance.**
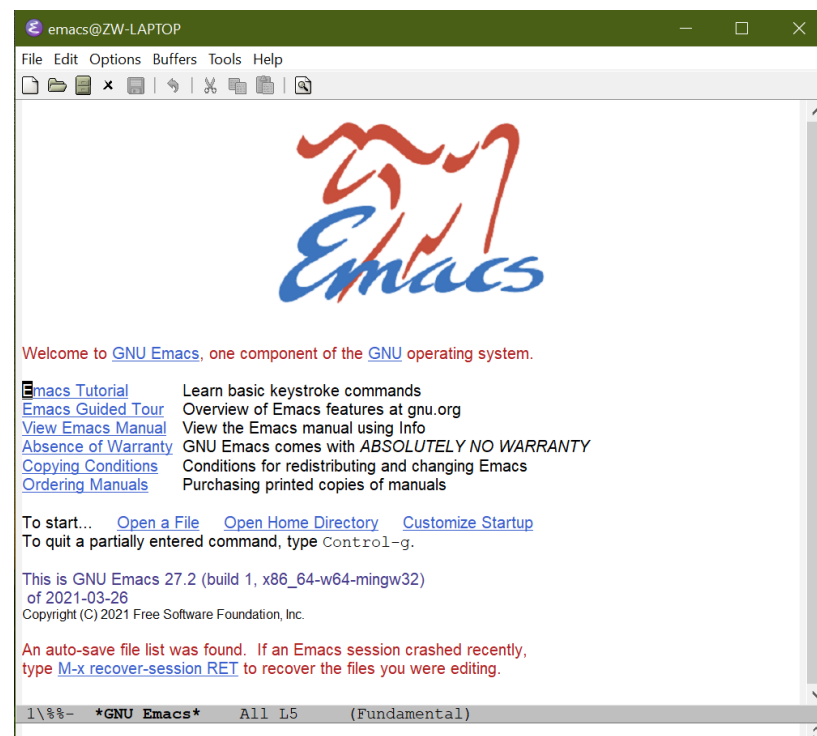
Raise your hand if you know someone who is afraid to experiment with their computer.

The left monitor displays a boot menu with the following options:

```
Debian GNU/Linux 12 (bookworm)
System Recovery (GRML)
Windows 10
Reboot Into Firmware Interface
```

Industry prioritizes "out of box experience"

- **Sleek, user friendly, ...**
  - No customization
  - No replaceable parts
  - Surveillance for marketing
- **Store everything in the cloud**
  - Can offload expensive computation to cloud
  - Expensive paperweight if the net is down
- ***Not programmable by end user***

WARNING!

SYSTEM MAY HAVE DETECTED VIRUSES ON YOUR COMPUTER

System May Have Found (2) Malicious Viruses: *Rootkit.Sirefef.Spy* and *Trojan.FakeAV-Download*. Your Personal & Financial Information **MAY NOT BE SAFE.**

For Help Removing Viruses, Call Tech Support Online Right Away:

1(855) 970-1892
(TOLL-FREE, High Priority Call Line)

Ads are the #1 vector for malware

https://www.crowdstrike.com/cybersecurity-101/malware/malvertising/
https://www.malwarebytes.com/malvertising

App stores are the #2 vector for malware
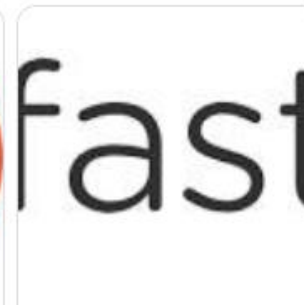

Turbo VPN - Secure VPN Proxy

SuperVPN Fast VPN Client

Thunder VPN - Fast, Safe VPN

Psiphon Pro

Fast VPN

SuperVPN Pro

Snap VPN

VPN.lat

LetsVPN

CandyLink VPN

ArgoVPN

VPN Pro

Not all problems have technical solutions

(Firefox + uBlock Origin kills malvertising dead)

(but then the newspaper goes out of business…)

You are Here → Political / Financial / Application / Presentation / Session / Transport / Network / Link / Physical

Evi Nemeth's extended OSI protocol stack, quoted in https://www.linuxjournal.com/article/10726
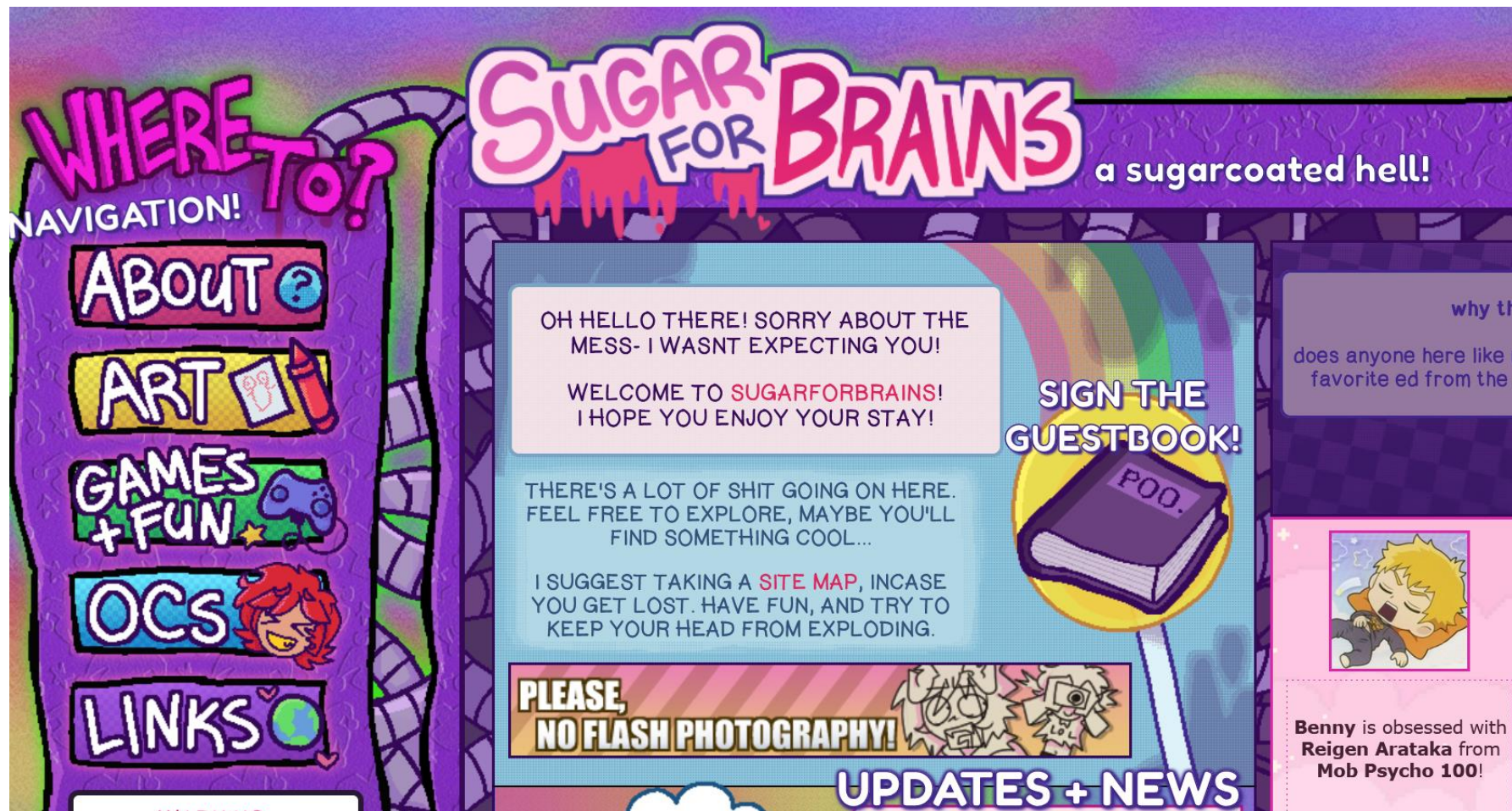
Another vision of the future

- **Modular**
  - Long design life
  - Repairable and upgradable
- **Cloud not required**
  - Works if network is down
  - You control your data
  - Lose device? Lose data
- **End user programmable**
  - Sacrifice polish and out-of-box experience

"If you depend on proprietary mainstream mobile messenger applications, banking applications, use loyalty or travel apps, consume DRM media, or play mobile video games on your fruit or Android smartphone, then the PinePhone Pro is likely not for you."

Why can't we have it both ways?

- **Seamless cloud backups AND offline operation?**

- **Easy to use out of box AND fully customizable and programmable?**

- **Arbitrary software installation AND malware-free app store?**

- **"Free software" operating system AND ability to run proprietary applications when necessary?**

- **Long service life, end user repairable?**
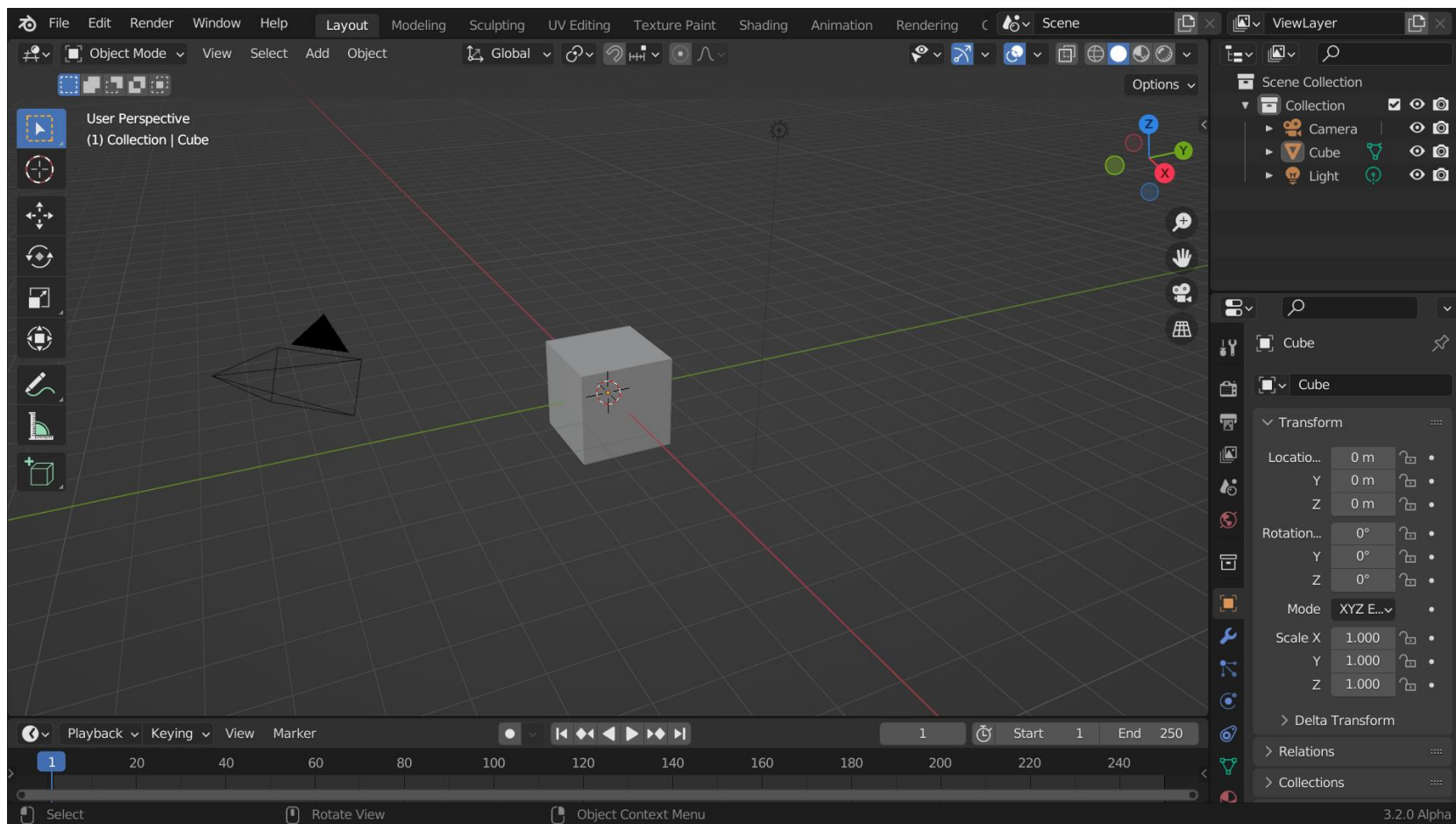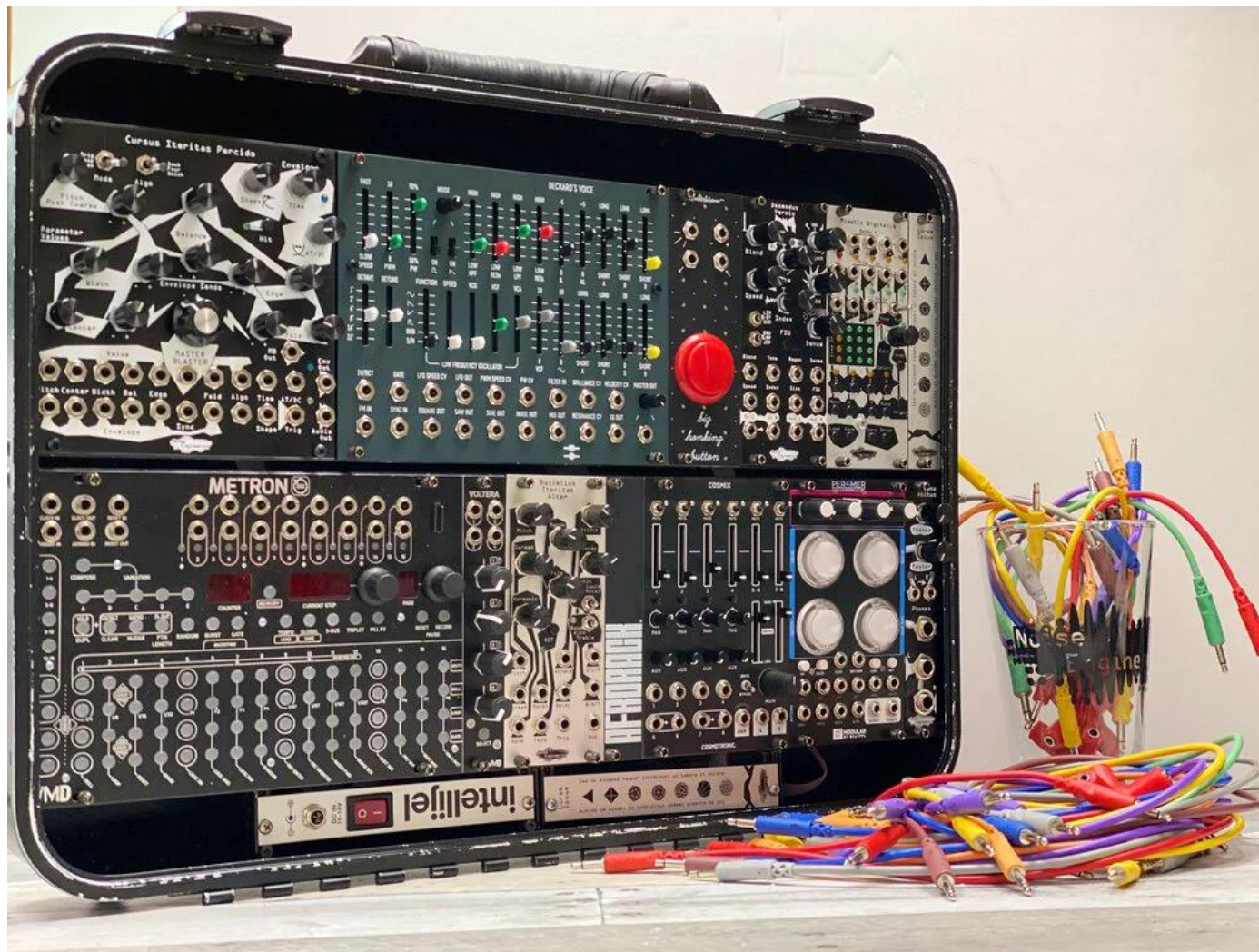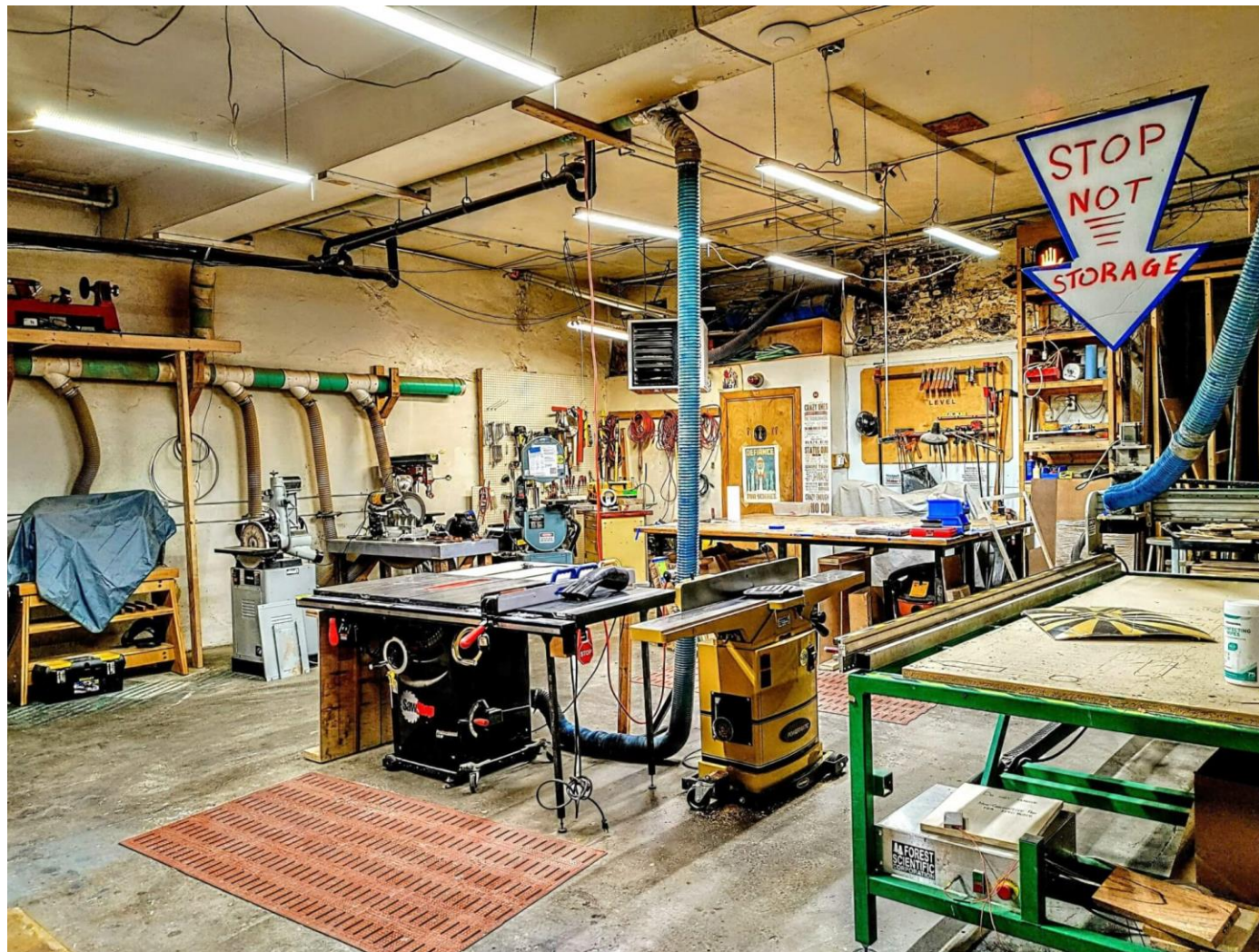
Customizability for Non-Programmers

https://sugarforbrains.neocities.org/

Customizability for Non-Programmers

https://www.youtube.com/watch?v=ccMwukZcv3k

Customizability for Non-Programmers

Customizability for Non-Programmers

https://noiseengineering.us/blogs/loquelic-literitas-the-blog/have-no-fear-modular-is-here-modular-synths-for-musicians

Customizability for Non-Programmers

https://hackpgh.org/

The Last Quiz

https://canvas.cmu.edu/courses/34989/quizzes/105559