Operating Systems

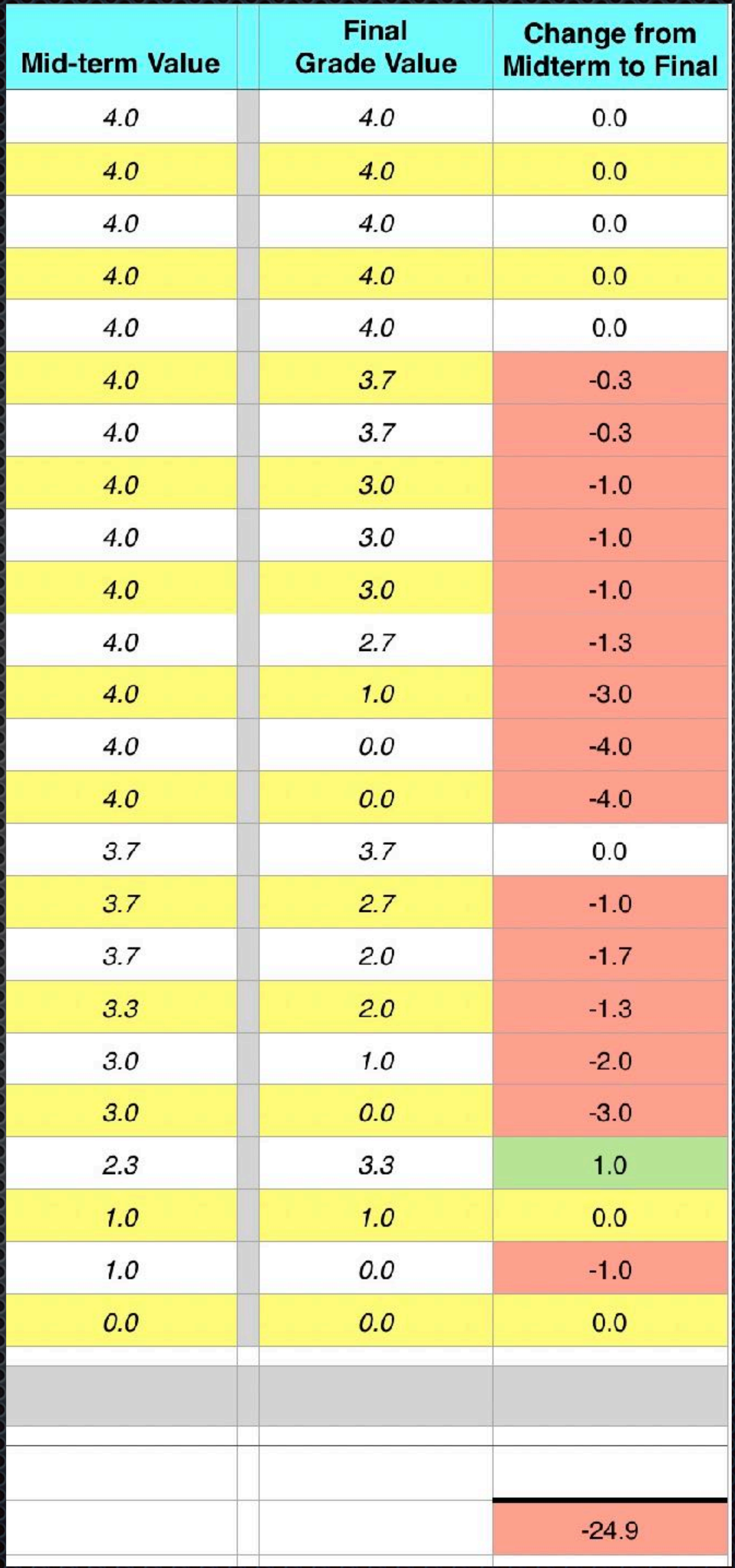
Some history and advice as we begin our adventure. . . .

Some Final Grades

A	*9*	A-	A	A	A	A	A	A	A	A
B	*6*	B-	B-	B-	B	B	B+			
C	*2*	C-	C+							
D	*6*	D	D	D	D	D	D			
F	*1*	F								
W	*4*	W	W	W	W					

There were no incompletes.
There won't be this time either. Don't even ask.

Some Midterm vs. Final Grades



Mid-term Value		Final Grade Value	Change from Midterm to Final
4.0		4.0	0.0
4.0		4.0	0.0
4.0		4.0	0.0
4.0		4.0	0.0
4.0		4.0	0.0
4.0		3.7	-0.3
4.0		3.7	-0.3
4.0		3.0	-1.0
4.0		3.0	-1.0
4.0		3.0	-1.0
4.0		2.7	-1.3
4.0		1.0	-3.0
4.0		0.0	-4.0
4.0		0.0	-4.0
3.7		3.7	0.0
3.7		2.7	-1.0
3.7		2.0	-1.7
3.3		2.0	-1.3
3.0		1.0	-2.0
3.0		0.0	-3.0
2.3		3.3	1.0
1.0		1.0	0.0
1.0		0.0	-1.0
0.0		0.0	0.0
			-24.9

Advice

- Attend every class. Don’t miss even one.
- Put your devices away and pay attention in class.
- Take notes in class (in pencil) on paper.
- Rewrite your class notes into a new notebook as a form of studying.
- Do every lab. Write them up in LaTeX.
- Work on your project every day. Commit to GitHub at least once a day if not more frequently.

Advice

- Attend every class. Don’t miss even one. Why?
  - If you’re not in class I won’t have the opportunity to invite you to join a faculty research project, or the IBM Joint Study, or a hackathon, or a CCAC project, or an ECRL project, or a Summer Fellowship, or the Marist Innovation Lab.
  - If you’re not in class I will never know you well enough to write you a letter of recommendation or recommend you to an employer.

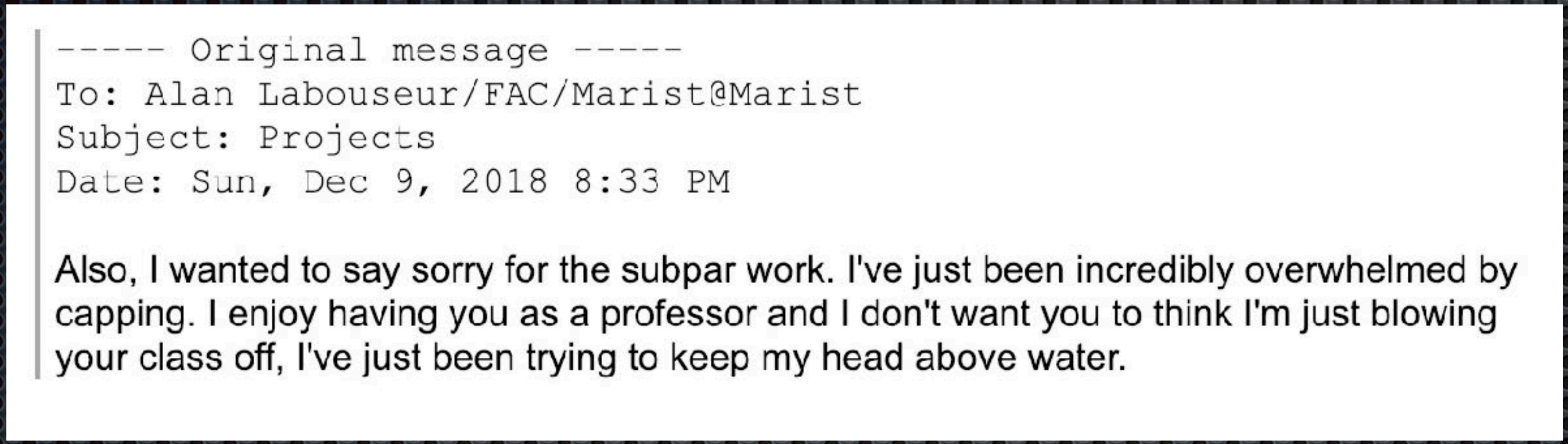
ADVICE from Dr. Helen Rothberg

- **A**ction — Do more, say less. Pursue your interests. Ask for help when you need it.
- **D**etermination — Stick to it and get things done.
- **V**ision — What are you trying to become? What do you want to do with your life? How will you get there?
- **I**ntegrity — Tell the truth all the time. Don't create or participate in drama.
- **C**ommunication — Be attentive and dig in, even if it's hard.
- **E**mpathy — Dare to care about yourself and others.

Do Not Make Excuses

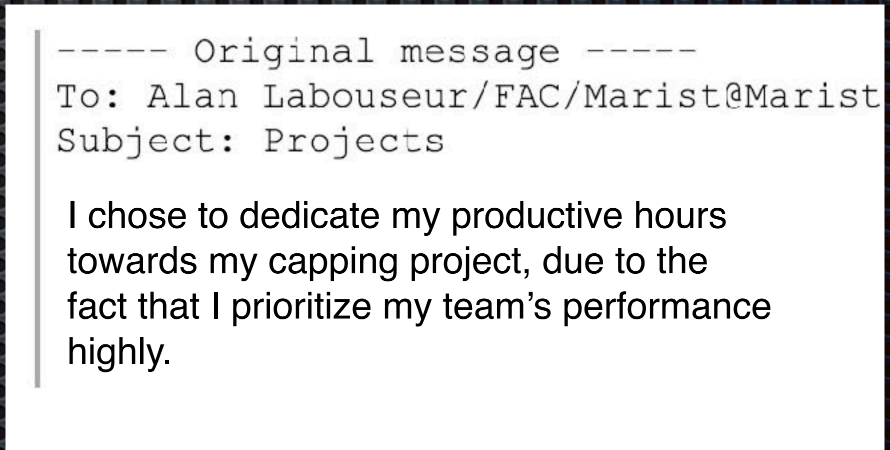
- I love Chris Algozzine, but I don’t give a damn about Capping.

- These people failed this class.



```
----- Original message -----
To: Alan Labouseur/FAC/Marist@Marist
Subject: Projects
Date: Sun, Dec 9, 2018 8:33 PM
```

Also, I wanted to say sorry for the subpar work. I've just been incredibly overwhelmed by capping. I enjoy having you as a professor and I don't want you to think I'm just blowing your class off, I've just been trying to keep my head above water.



```
----- Original message -----
To: Alan Labouseur/FAC/Marist@Marist
Subject: Projects
```

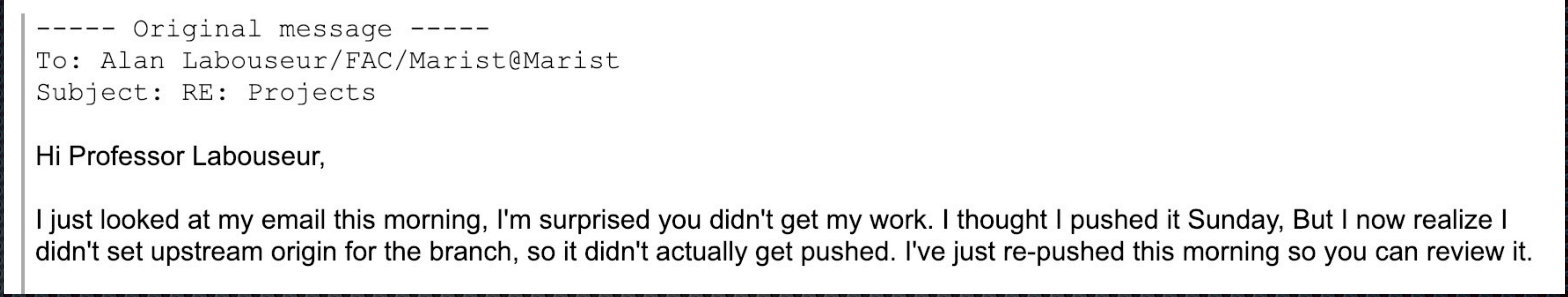
I chose to dedicate my productive hours towards my capping project, due to the fact that I prioritize my team’s performance highly.

- Work on your operating system project **every** day.

- Don’t fall behind in this class (or in Capping).

Do Not Make Excuses

- This is not acceptable:



```
----- Original message -----
To: Alan Labouseur/FAC/Marist@Marist
Subject: RE: Projects
```

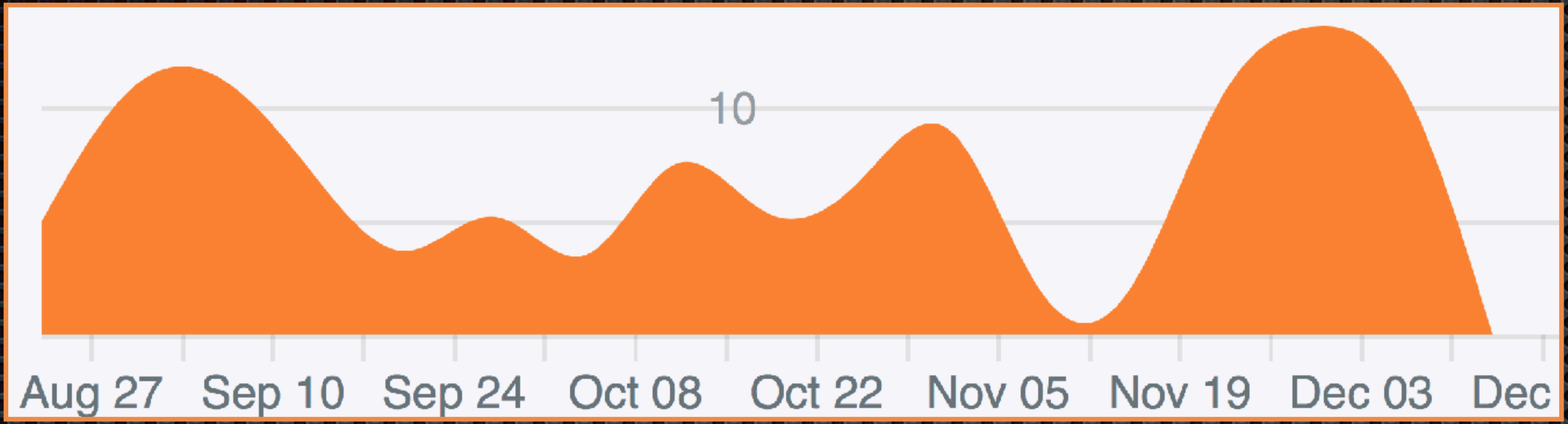
Hi Professor Labouseur,

I just looked at my email this morning, I'm surprised you didn't get my work. I thought I pushed it Sunday, But I now realize I didn't set upstream origin for the branch, so it didn't actually get pushed. I've just re-pushed this morning so you can review it.

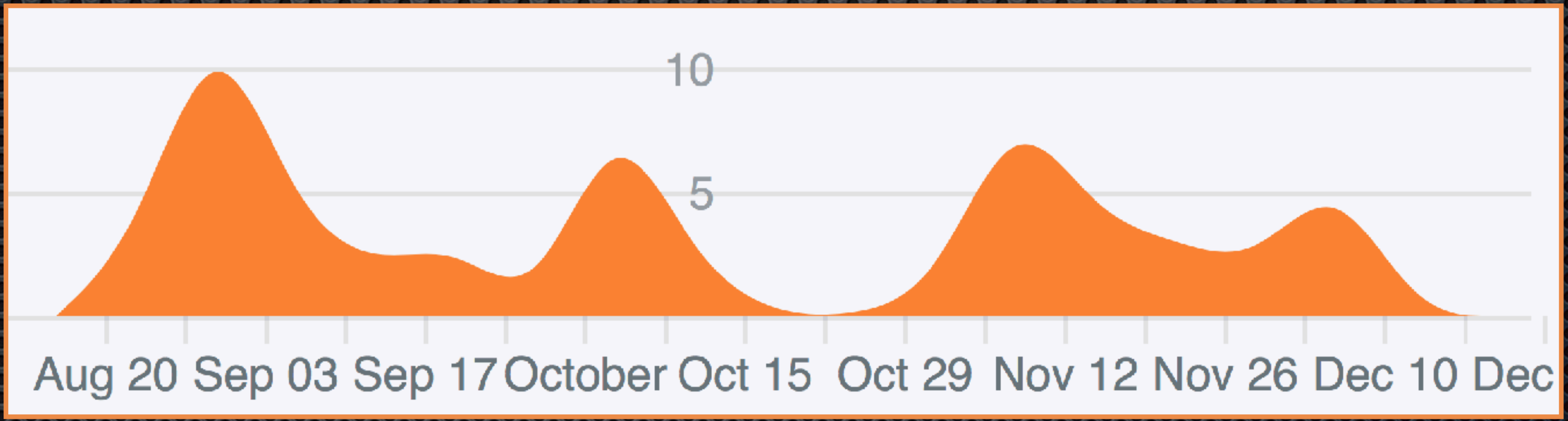
- I did not “review it”. The grade remained the same.
- If you cannot figure out GitHub, you will not be earning a Computer Science degree.
- Speaking of GitHub . . .

GitHub Commit Graphs

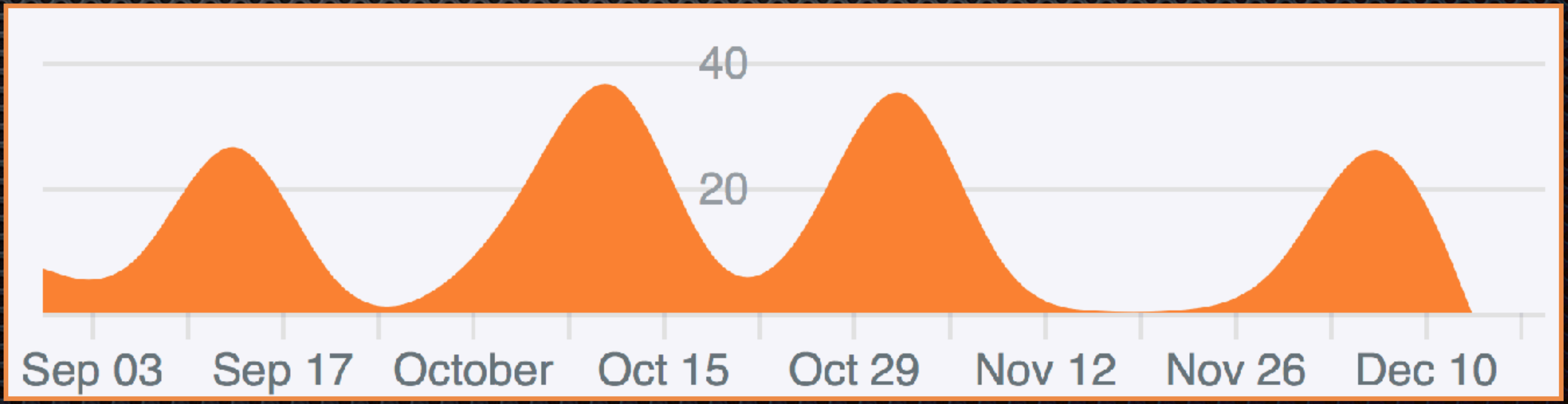
A



A-

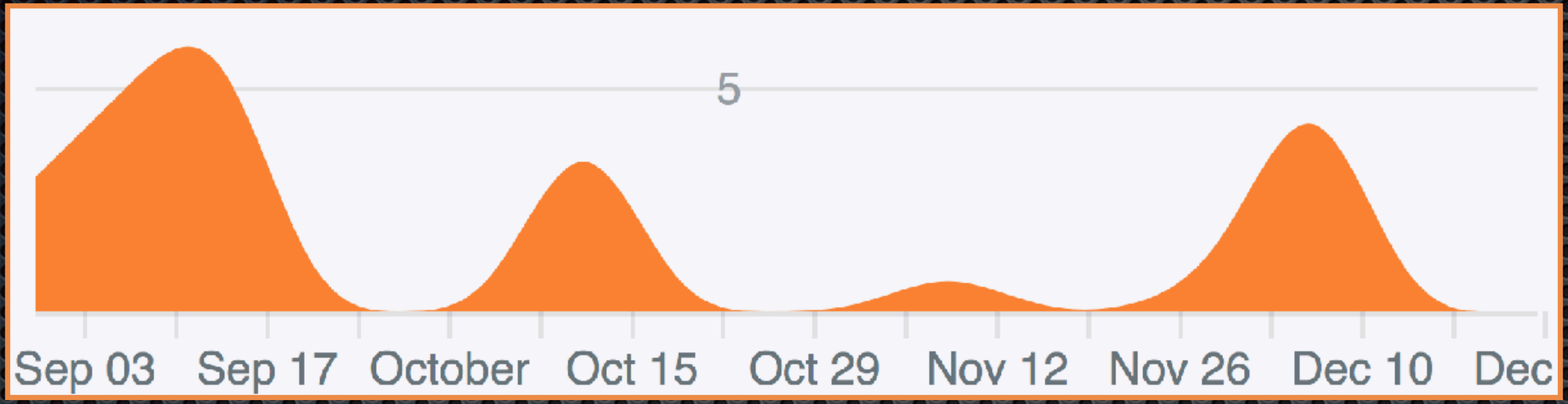


B

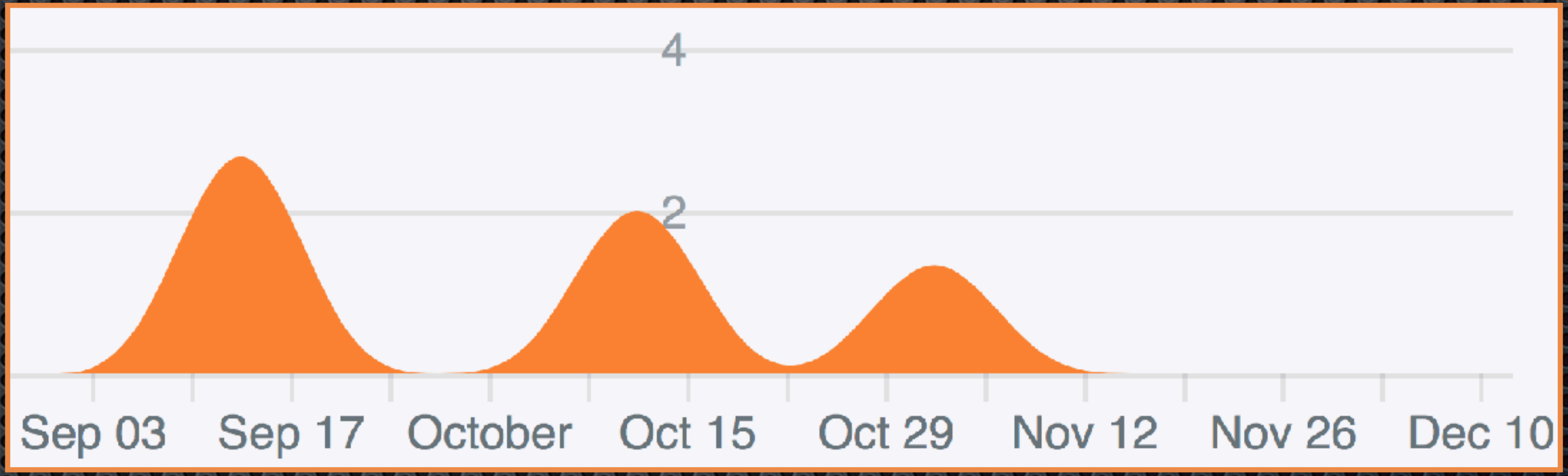


GitHub Commit Graphs

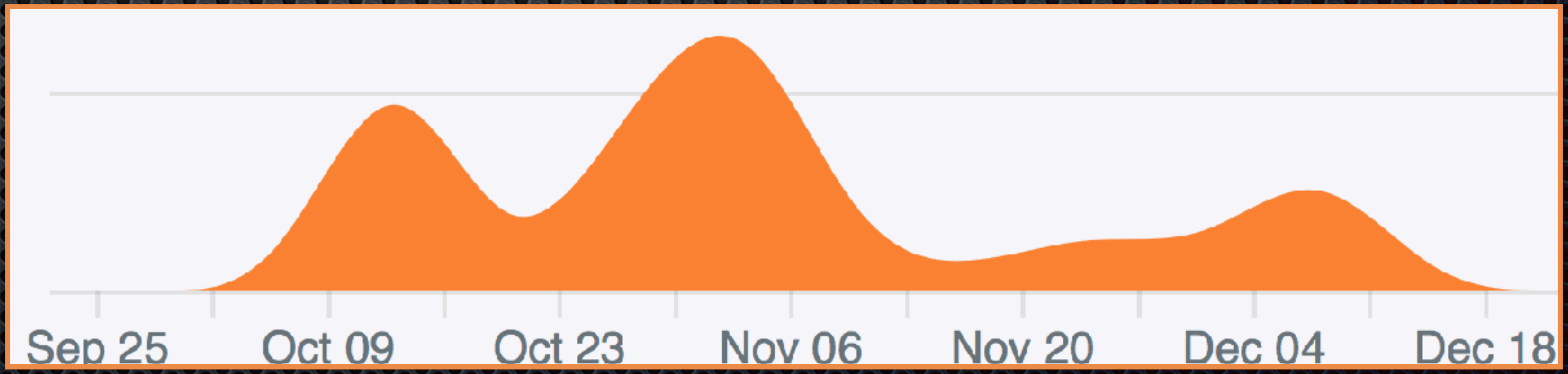
C



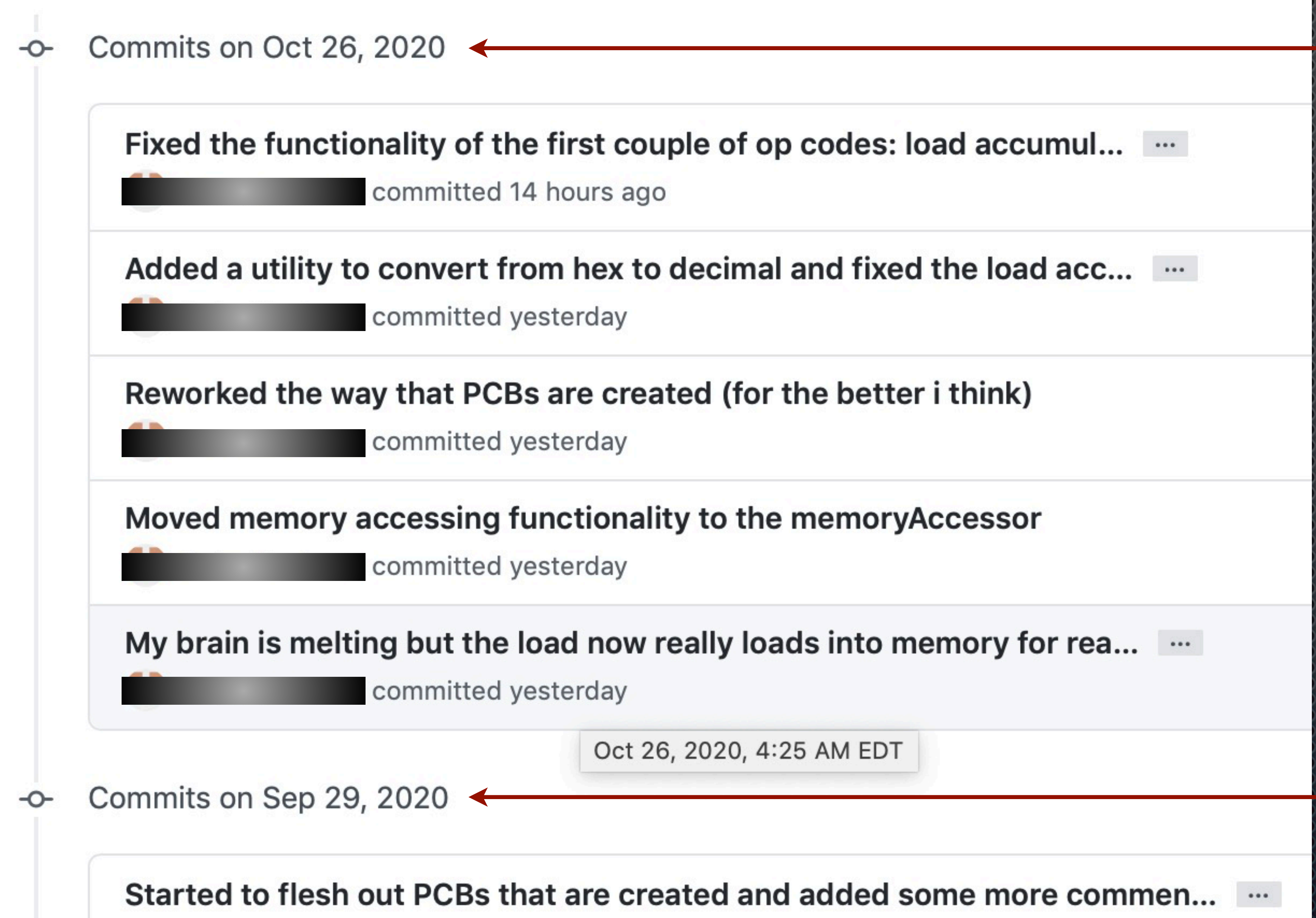
D



F



GitHub Commits



Commits on Oct 26, 2020

Fixed the functionality of the first couple of op codes: load accumul... ···
[redacted] committed 14 hours ago

Added a utility to convert from hex to decimal and fixed the load acc... ···
[redacted] committed yesterday

Reworked the way that PCBs are created (for the better i think)
[redacted] committed yesterday

Moved memory accessing functionality to the memoryAccessor
[redacted] committed yesterday

My brain is melting but the load now really loads into memory for rea... ···
[redacted] committed yesterday

Oct 26, 2020, 4:25 AM EDT

Commits on Sep 29, 2020

Started to flesh out PCBs that are created and added some more commen... ···

Project 3 due date

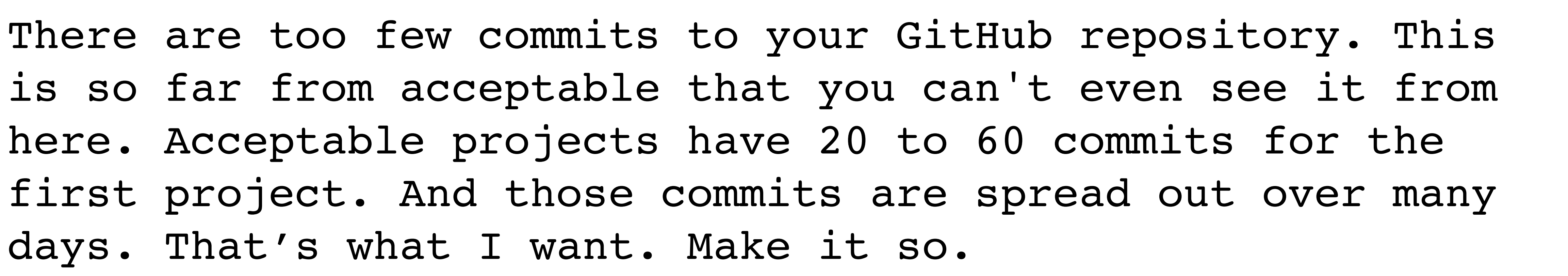
No activity.

Project 2 due date +1 day

GitHub Commits

- You must commit frequently over many days as you work on our project. You should have over 100 commits when you complete the final project

- This is an actual e-mail I sent to a student regarding



> There are too few commits to your GitHub repository. This is so far from acceptable that you can't even see it from here. Acceptable projects have 20 to 60 commits for the first project. And those commits are spread out over many days. That’s what I want. Make it so.

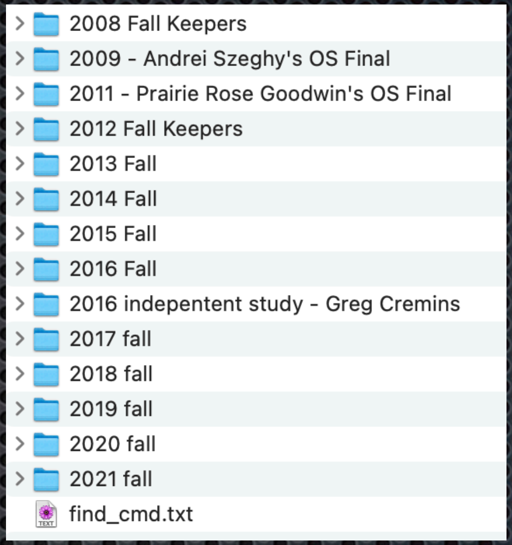
- Do not be the next student to get this e-mail from me.

Academic Honesty

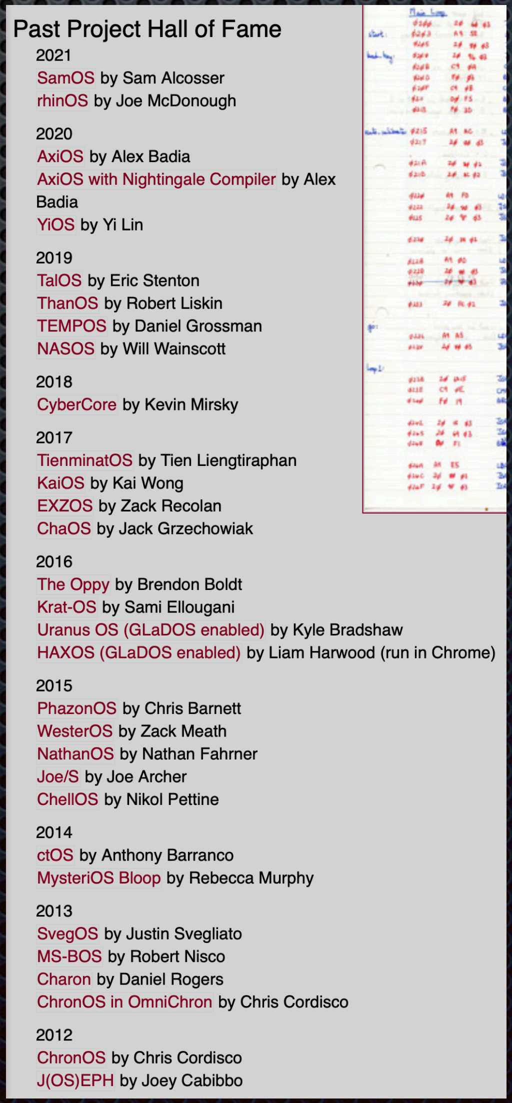
- These are individual projects.
  All work must be your own.
- Any violation of this will result in your **immediate** and **automatic** failure of this class.
- This is your only warning. The next time we talk about this it will be in the context of me telling you that you have failed this class.
- I am serious. Ask prior students.

Academic Honesty

- I will be checking all your code against...
  - your class
  - other classes
  - all past projects
  - the Hall of Fame
  - and...



2008 Fall Keepers
2009 - Andrei Szeghy's OS Final
2011 - Prairie Rose Goodwin's OS Final
2012 Fall Keepers
2013 Fall
2014 Fall
2015 Fall
2016 Fall
2016 indepentent study - Greg Cremins
2017 fall
2018 fall
2019 fall
2020 fall
2021 fall
find_cmd.txt



Past Project Hall of Fame

2021
SamOS by Sam Alcosser
rhinOS by Joe McDonough

2020
AxiOS by Alex Badia
AxiOS with Nightingale Compiler by Alex Badia
YiOS by Yi Lin

2019
TalOS by Eric Stenton
ThanOS by Robert Liskin
TEMPOS by Daniel Grossman
NASOS by Will Wainscott

2018
CyberCore by Kevin Mirsky

2017
TienminatOS by Tien Liengtiraphan
KaiOS by Kai Wong
EXZOS by Zack Recolan
ChaOS by Jack Grzechowiak

2016
The Oppy by Brendon Boldt
Krat-OS by Sami Ellougani
Uranus OS (GLaDOS enabled) by Kyle Bradshaw
HAXOS (GLaDOS enabled) by Liam Harwood (run in Chrome)

2015
PhazonOS by Chris Barnett
WesterOS by Zack Meath
NathanOS by Nathan Fahrner
Joe/S by Joe Archer
ChellOS by Nikol Pettine

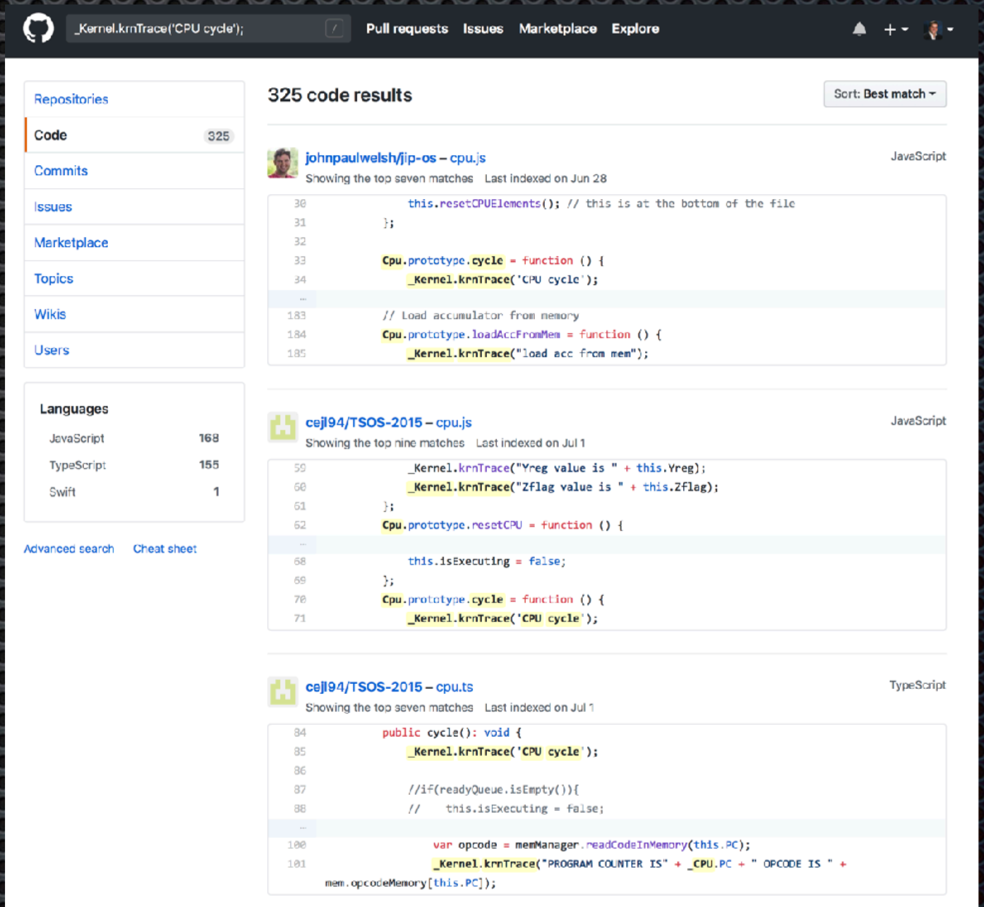
2014
ctOS by Anthony Barranco
MysteriOS Bloop by Rebecca Murphy

2013
SvegOS by Justin Svegliato
MS-BOS by Robert Nisco
Charon by Daniel Rogers
ChronOS in OmniChron by Chris Cordisco

2012
ChronOS by Chris Cordisco
J(OS)EPH by Joey Cabibbo

Academic Honesty

- I will be checking all your code against…
  - your class
  - other classes
  - all past projects
  - the Hall of Fame
  - and…
  - GitHub



325 code results

Languages	
JavaScript	168
TypeScript	155
Swift	1

johnpaulwelsh/jip-os – cpu.js — JavaScript
Showing the top seven matches   Last indexed on Jun 28

```
            this.resetCPUElements(); // this is at the bottom of the file
        };

        Cpu.prototype.cycle = function () {
            _Kernel.krnTrace('CPU cycle');
…
        // Load accumulator from memory
        Cpu.prototype.loadAccFromMem = function () {
            _Kernel.krnTrace("load acc from mem");
```

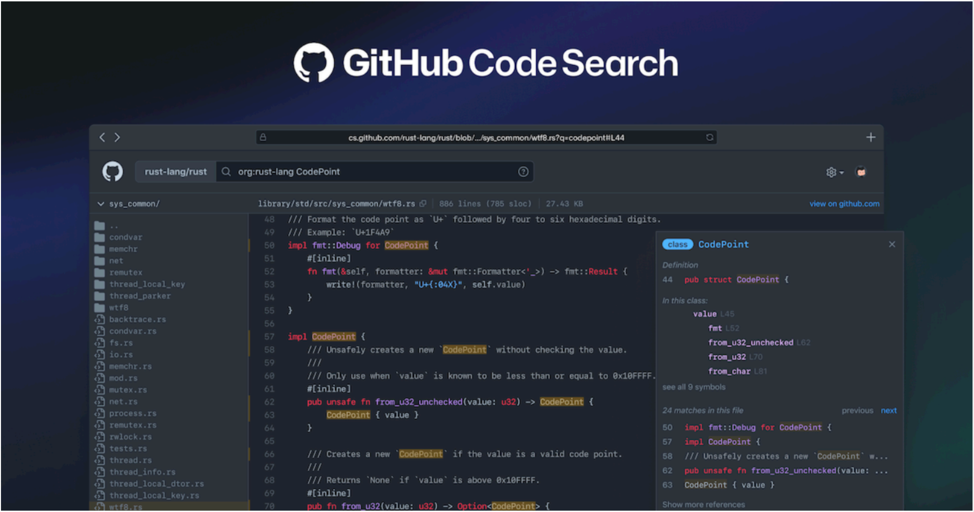
cejl94/TSOS-2015 – cpu.js — JavaScript
Showing the top nine matches   Last indexed on Jul 1

```
            _Kernel.krnTrace("Yreg value is " + this.Yreg);
            _Kernel.krnTrace("Zflag value is " + this.Zflag);
        };
        Cpu.prototype.resetCPU = function () {
…
            this.isExecuting = false;
        };
        Cpu.prototype.cycle = function () {
            _Kernel.krnTrace('CPU cycle');
```

cejl94/TSOS-2015 – cpu.ts — TypeScript
Showing the top seven matches   Last indexed on Jul 1

```
        public cycle(): void {
            _Kernel.krnTrace('CPU cycle');

            //if(readyQueue.isEmpty()){
            //    this.isExecuting = false;
…
                var opcode = memManager.readCodeInMemory(this.PC);
                _Kernel.krnTrace("PROGRAM COUNTER IS" + _CPU.PC + " OPCODE IS " +
    mem.opcodeMemory[this.PC]);
```



library/std/src/sys_common/wtf8.rs | 886 lines (785 sloc) | 27.43 KB

```
/// Format the code point as `U+` followed by four to six hexadecimal digits.
/// Example: `U+1F4A9`
impl fmt::Debug for CodePoint {
    #[inline]
    fn fmt(&self, formatter: &mut fmt::Formatter<'_>) -> fmt::Result {
        write!(formatter, "U+{:04X}", self.value)
    }
}

impl CodePoint {
    /// Unsafely creates a new `CodePoint` without checking the value.
    ///
    /// Only use when `value` is known to be less than or equal to 0x10FFFF.
    #[inline]
    pub unsafe fn from_u32_unchecked(value: u32) -> CodePoint {
        CodePoint { value }
    }

    /// Creates a new `CodePoint` if the value is a valid code point.
    ///
    /// Returns `None` if `value` is above 0x10FFFF.
    #[inline]
    pub fn from_u32(value: u32) -> Option<CodePoint> {
```

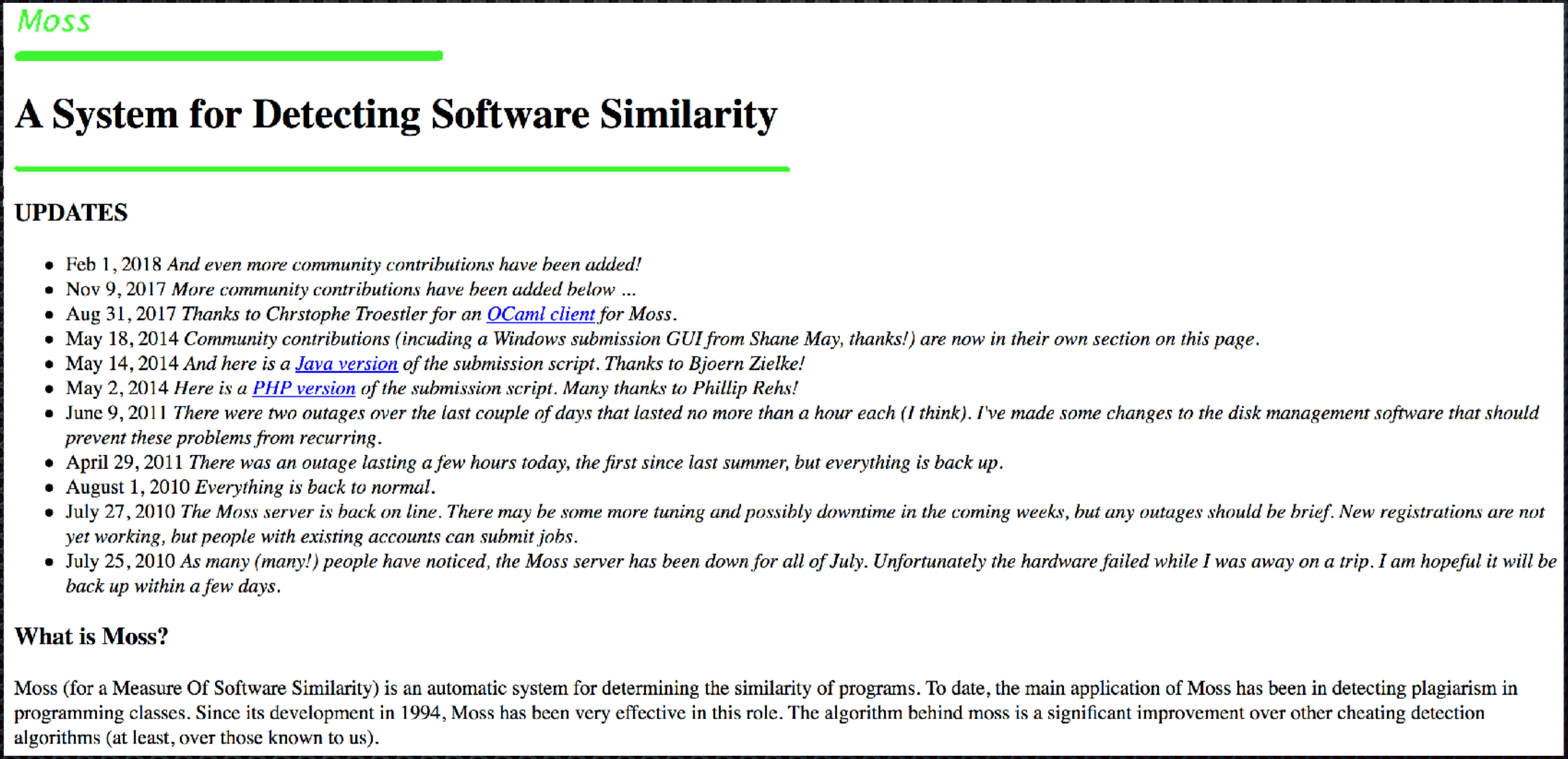
```
44 pub struct CodePoint {
```

```
value L45
    fmt L52
    from_u32_unchecked L62
    from_u32 L70
    from_char L81
```

```
50 impl fmt::Debug for CodePoint {
57 impl CodePoint {
58 /// Unsafely creates a new `CodePoint` w...
62 pub unsafe fn from_u32_unchecked(value: ...
63 CodePoint { value }
```

Academic Honesty

- I will be using MOSS to help detect cheaters. See theory.stanford.edu/~aiken/moss/



Moss

A System for Detecting Software Similarity

UPDATES

- Feb 1, 2018 *And even more community contributions have been added!*
- Nov 9, 2017 *More community contributions have been added below ...*
- Aug 31, 2017 *Thanks to Chrstophe Troestler for an OCaml client for Moss.*
- May 18, 2014 *Community contributions (incuding a Windows submission GUI from Shane May, thanks!) are now in their own section on this page.*
- May 14, 2014 *And here is a Java version of the submission script. Thanks to Bjoern Zielke!*
- May 2, 2014 *Here is a PHP version of the submission script. Many thanks to Phillip Rehs!*
- June 9, 2011 *There were two outages over the last couple of days that lasted no more than a hour each (I think). I've made some changes to the disk management software that should prevent these problems from recurring.*
- April 29, 2011 *There was an outage lasting a few hours today, the first since last summer, but everything is back up.*
- August 1, 2010 *Everything is back to normal.*
- July 27, 2010 *The Moss server is back on line. There may be some more tuning and possibly downtime in the coming weeks, but any outages should be brief. New registrations are not yet working, but people with existing accounts can submit jobs.*
- July 25, 2010 *As many (many!) people have noticed, the Moss server has been down for all of July. Unfortunately the hardware failed while I was away on a trip. I am hopeful it will be back up within a few days.*

What is Moss?

Moss (for a Measure Of Software Similarity) is an automatic system for determining the similarity of programs. To date, the main application of Moss has been in detecting plagiarism in programming classes. Since its development in 1994, Moss has been very effective in this role. The algorithm behind moss is a significant improvement over other cheating detection algorithms (at least, over those known to us).

Write Original Code

- Infuse your code with your personality.
  - variable names, code conventions, and style
  - structure and modularization
- Write grammatically correct comments and steep them with your personality. Be snarky, clever, and funny if you like. I want to get to know you by reading your code.
  - Comments should say things about the code that the code cannot say about itself.
- List collaborators (including AI), references, and sites consulted in a comment block at the top of each file.

What about using an AI?

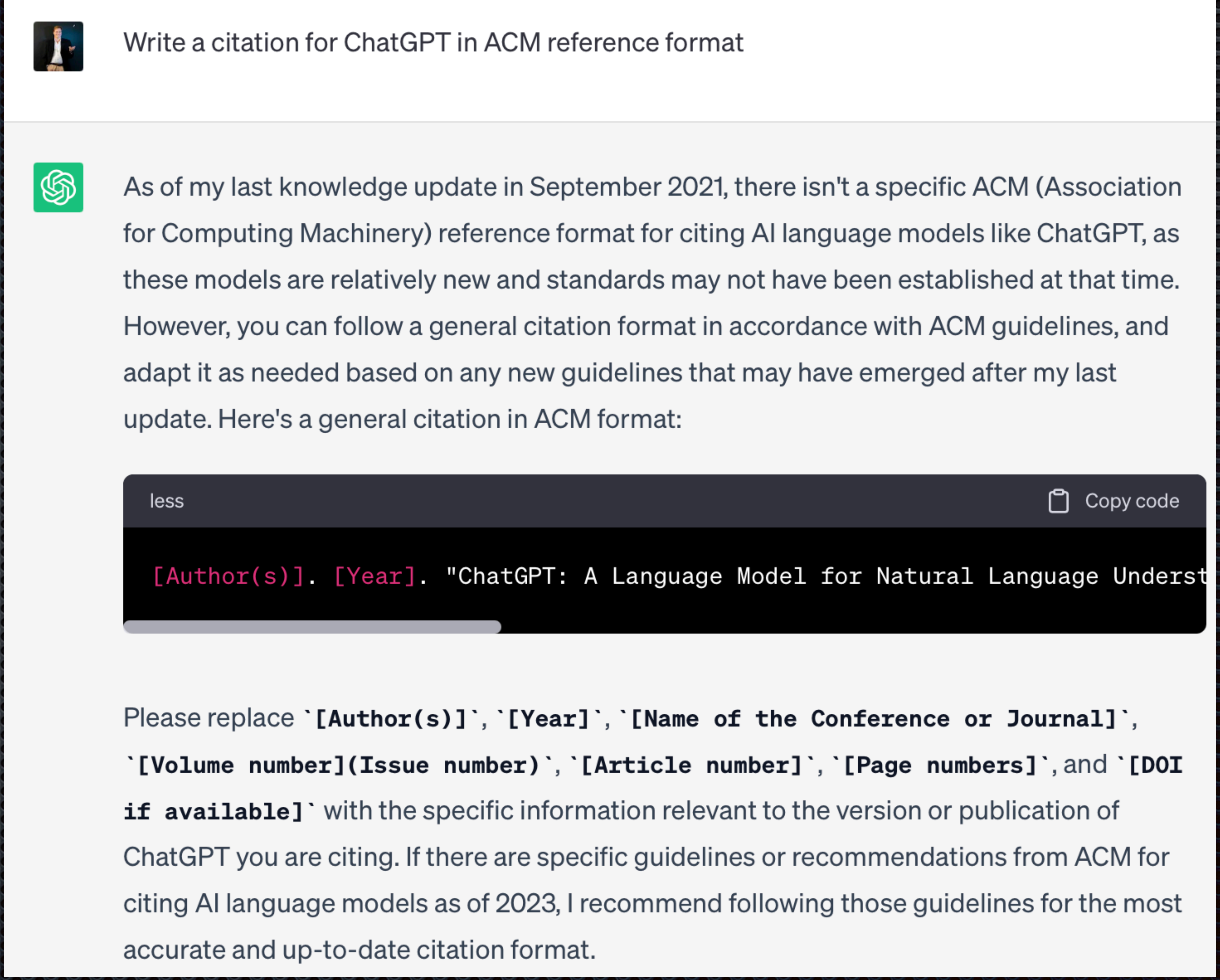
- I've been teaching my teenage son to drive. We started with golf carts and then moved to a real car in empty parking lots.
- He's making incremental progress built on a solid foundation (and cart paths). But he's a long way from driving in F1.
- If I were to put him in an Aston Martin he'd surely hurt himself and others.
- AI code generators are like F1 racecars. Until you have a solid foundation you're only going to hurt yourself and others by using them. But if you feel that you’re ready… DRS is enabled so go for it!

When using an AI

- As ever, anything that’s not 100% yours and yours alone **must be marked and cited with sources**. Whether you get help from an online AI, a tutor in the programming lab, or a random student sleeping outside of Marion Hall, you have to acknowledge that help and cite it or it is unquestionably plagiarism. Copying and pasting from a Hall of Fame project or GitHub or some other online source is unquestionably plagiarism unless you cite it; and this is no different.

- How do you cite it? Let’s ask ChatGPT . . .

When using an AI



Write a citation for ChatGPT in ACM reference format

As of my last knowledge update in September 2021, there isn't a specific ACM (Association for Computing Machinery) reference format for citing AI language models like ChatGPT, as these models are relatively new and standards may not have been established at that time. However, you can follow a general citation format in accordance with ACM guidelines, and adapt it as needed based on any new guidelines that may have emerged after my last update. Here's a general citation in ACM format:

```
[Author(s)]. [Year]. "ChatGPT: A Language Model for Natural Language Underst
```

Please replace `[Author(s)]`, `[Year]`, `[Name of the Conference or Journal]`, `[Volume number](Issue number)`, `[Article number]`, `[Page numbers]`, and `[DOI if available]` with the specific information relevant to the version or publication of ChatGPT you are citing. If there are specific guidelines or recommendations from ACM for citing AI language models as of 2023, I recommend following those guidelines for the most accurate and up-to-date citation format.

When using an AI

- I’m interested in learning about how well (or not?) AI-assisted software development works for you. If you make use of an AI you must document your experience with it for every project: the chat interactions, what worked and what didn’t, challenges, unexpected occurrences, etc. Include a reflection about the experience as well. I’m interested to learn from you and your experiences with this new technology.

- Remember: plagiarism is one of the few sins where it is easier to receive permission than forgiveness. So if you are at all unsure about any aspect of this, ask me about it first.

Before asking me for help

- Review your code yourself by reading it from the perspective of someone seeing it for the first time.
    - Consider your variable names — are they descriptive?
    - Did you write plentiful comments about why you made those choices rather than what the code does?
    - Examine your code conventions and style — is it consistent?
    - Fix anything that’s missing, unclear, or confusing.

Before asking me for help

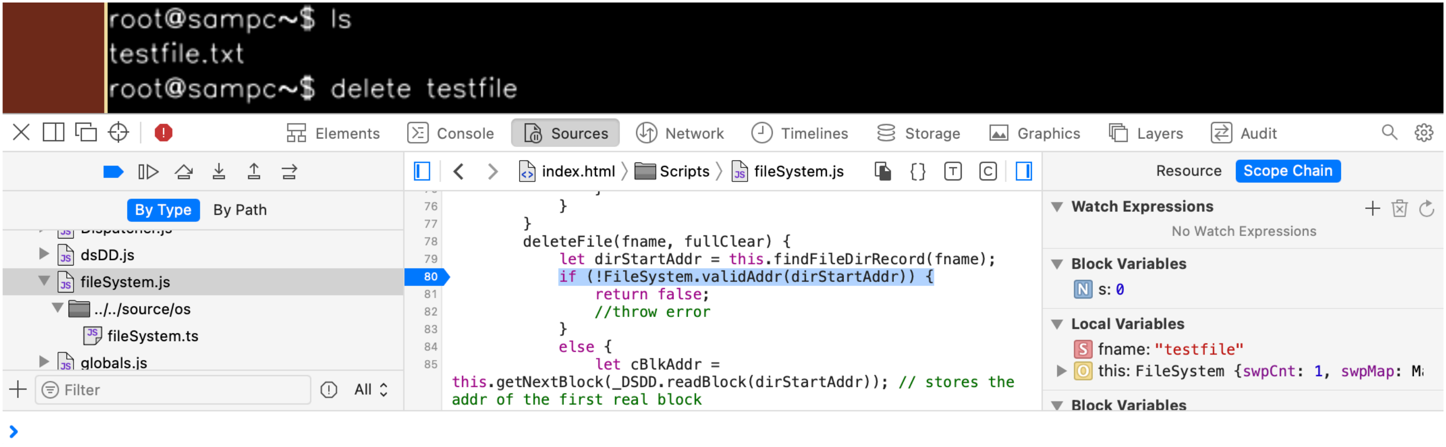
- Set a break point in your browser or IDE and step through your code line by line for several test cases.
    - This is the first thing I'm going to ask you to do, so you might as well do it preemptively.
    - You will be **amazed** at how many

      (1) mistakes,

      (2) bugs, and

      (2) off-by-one errors

      you'll find.



```
root@sampc~$ ls
testfile.txt
root@sampc~$ delete testfile
```

```
            }
        }
        deleteFile(fname, fullClear) {
            let dirStartAddr = this.findFileDirRecord(fname);
            if (!FileSystem.validAddr(dirStartAddr)) {
                return false;
                //throw error
            }
            else {
                let cBlkAddr =
this.getNextBlock(_DSDD.readBlock(dirStartAddr)); // stores the
addr of the first real block
```

Watch Expressions
No Watch Expressions

Block Variables
N s: 0

Local Variables
S fname: "testfile"
O this: FileSystem {swpCnt: 1, swpMap: M

Block Variables

When asking me for help

- Explain the problem clearly. Context matters.
- Describe several test cases that illustrate the problem.
- Tell me what you observed when stepping through your code line by line. (Do not ask me for help unless you have already done this. Seriously.)
- Make sure the latest version is committed on GitHub and include a link to it in your e-mail.