



# Computer Systems and C Programming

Prof. Darrell Long  
CSE 13S

*When the going gets weird, the  
weird turn pro.*  
—Dr. Hunter S. Thompson

29 March 2021

© 2021 Darrell Long

1

# This is a hard class

*That fact is not meant to scare you*

---

This course is *foundational*.

What you learn will be used from now on.

It is 7 Units, combining the old CMPE 13 and CMPE 13L.

You must pass all laboratory components to pass the course.



**KEEP  
CALM  
AND  
BUCKLE  
UP**



# What am I expected to know?

- High school algebra:
  - Ideally, you know a little Calculus.
- How to compose a paragraph in English.
- Some exposure to programming, for example, *Python*.
- The material from CSE 12/L:
  - Binary Arithmetic
  - Memory
  - Registers
  - Pointers
  - Instructions
  - Addressing modes



John Long  
*Video Game Test Engineer*



# Grading

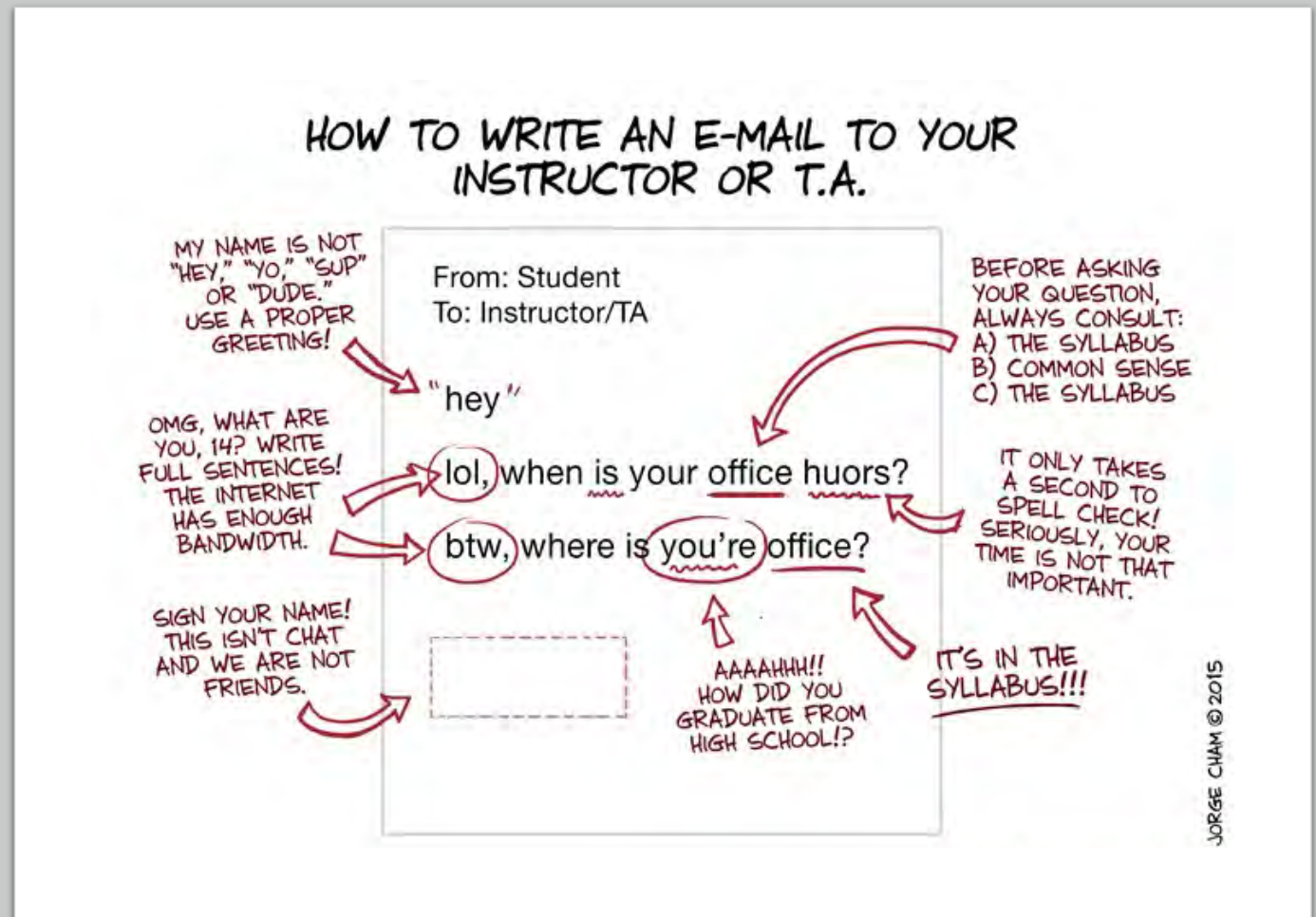
- Assignments: 50%
  - Examinations: 30%
  - Quizzes: 20%
- 
- We do not have the resources to grade extra credit.
  - Be sure that you do *each and every assignment*.
    - You must pass *every* assignment in order to pass the class.

## GRADING RUBRIC

PROBLEM 1 (TOTAL POINTS: 10)



Be nice to  
the TA



# DRC Accommodations

- Sabrina Au (TA) will act as coordinator.
- We can accommodate your testing requirements.
- We can offer *at most 2 extras days* for programs.
  - 2× on tests does not mean double time for programs.
- If you are going to be late, you must tell us *2 days* in advance!

29 March 2021



© 2021 Darrell Long



# Forms of Address

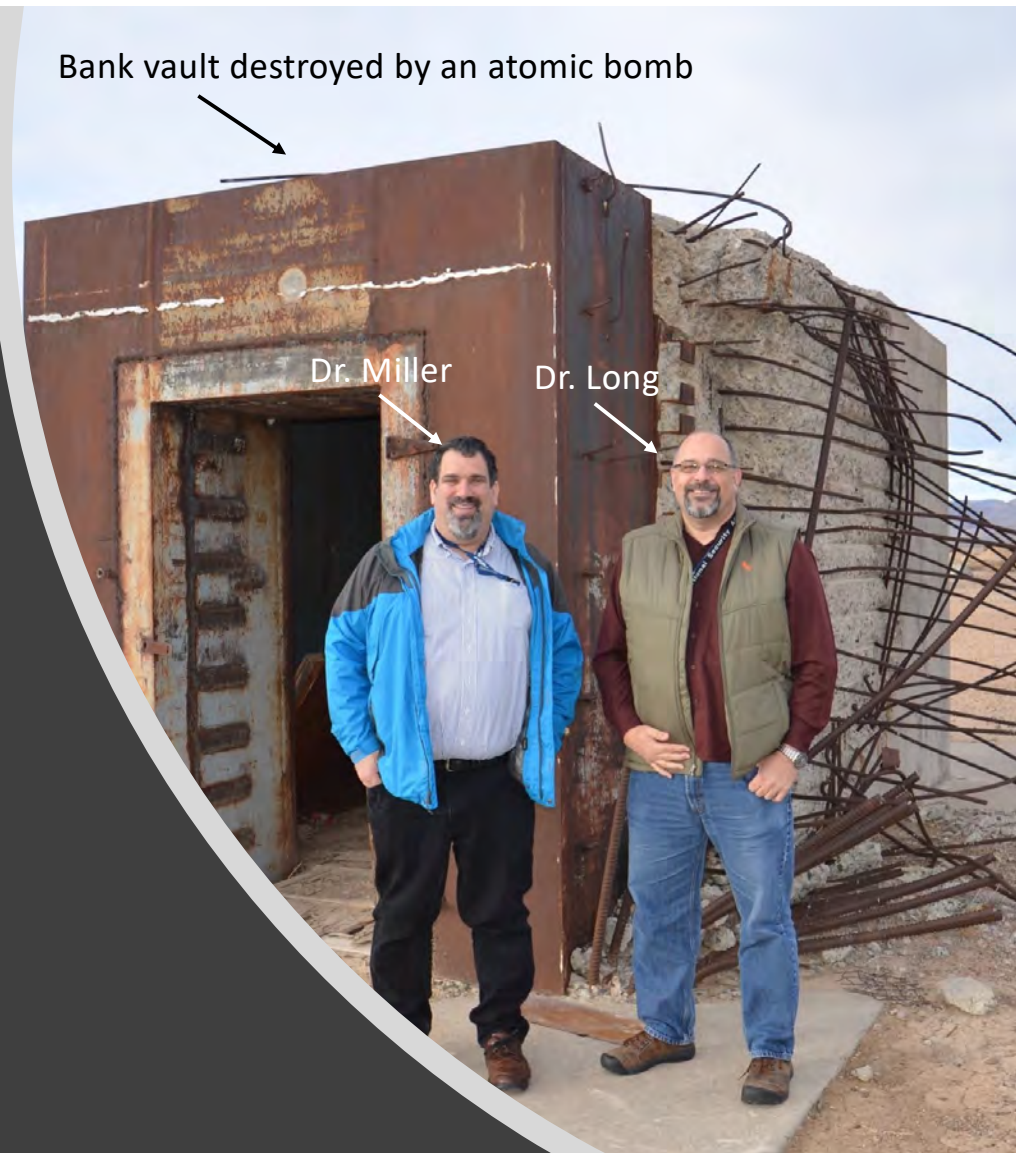
You can call me:

- Dr. Long
- Prof. Long
- Monsieur le Professeur Long

*No one* calls me by my first name, except my wife.

I will address you in any manner that you prefer, so long as it is polite.

- If I forget, just remind me.
- I once had a student named *Superman*. Yes, it was his real name.





# You will write a lot of programs in C



You may have seen Java or Python, but now it's time to learn C.

You will:

- ... need to worry about memory,
- ... use pointers,
- ... manipulate bits,
- ... use an actual editor,
- ... use UNIX (Linux), and
- ... use tools like *git*, *vim*, *make*, *cc*, *lldb*, etc.



Sir James Augustus Henry Murray, LL.D.

29 March 2021

# You will be busy

---

- Do not think that you can do it the night before...
  - Do not think you can do it the day before that...
  - In other words, start on it early.
- Programming requires *practice*.
- It requires *thought*.
- It requires the willingness to *rewrite* your code.

© 2021 Darrell Long

10

# You will get your hands dirty!

This class will show you where the bits hit the silicon.

You will write most of the code that you use:

- In contrast to some classes which teach you which libraries to use and give you code templates to fill in.

Ever wonder how a Python dictionary works?

- You're going to find out, it's called a hash table.

How about those lists in Python?

- We will look at arrays as well as linked lists.



Mike Rowe  
A Personal Hero





Necronomicon

# Reading

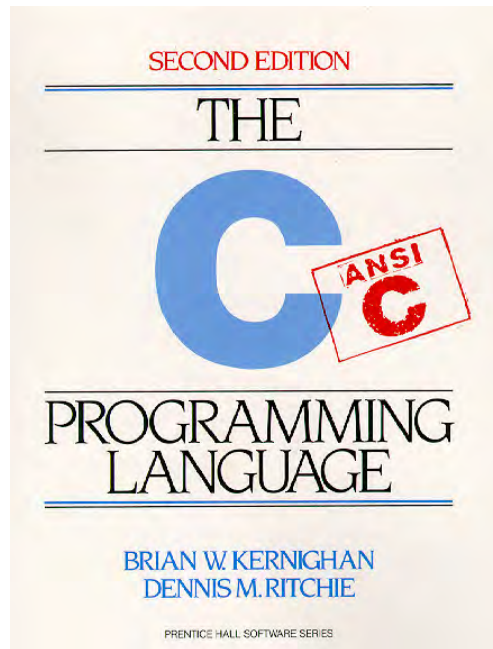
So many students have suffered for failing to read the book...

In order to succeed in your classes you must read the material.

- You cannot pass this course by simply:
- ... listening to lectures,
- ... reading these slides, or
- ... watching YouTube videos.

Read the appropriate section of the relevant book *before* you ask me, the TA, or the Tutor for help.

- If you have not done your part and completed the reading, you will go to the back of the line so that we can help students who have done the reading.



29 March 2021

# The C Programming Language

- The beauty and the danger of **C** is its simplicity.
  - **C** gives you more access to the raw machine than any language other than assembly language.
- **C** does not hold your hand. It will not protect you from your mistakes.
  - **C** requires you to think about what you are doing.
  - **C** will do exactly what you ask of it.
- It is rather like giving a monkey a chainsaw.

© 2021 Darrell Long

13

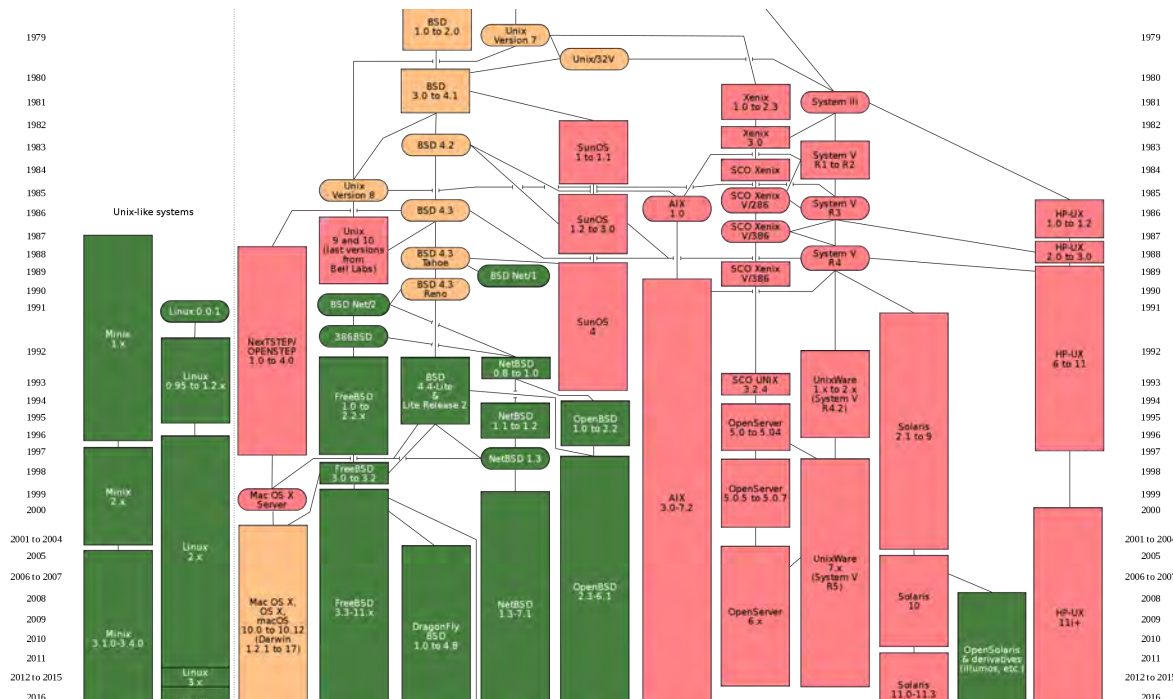
# It's about learning how to program

- Programming is not simply learning C, or Java, or Python.
  - It is learning how to think!
- There is a lot of difference between an adequate programmer and a good programmer,
- And even more difference between a good program and a great programmer.
- Get the book, it's free to you on O'Reilly!

The difference between theory and practice is greater in practice than in theory. —*John Wilkes*

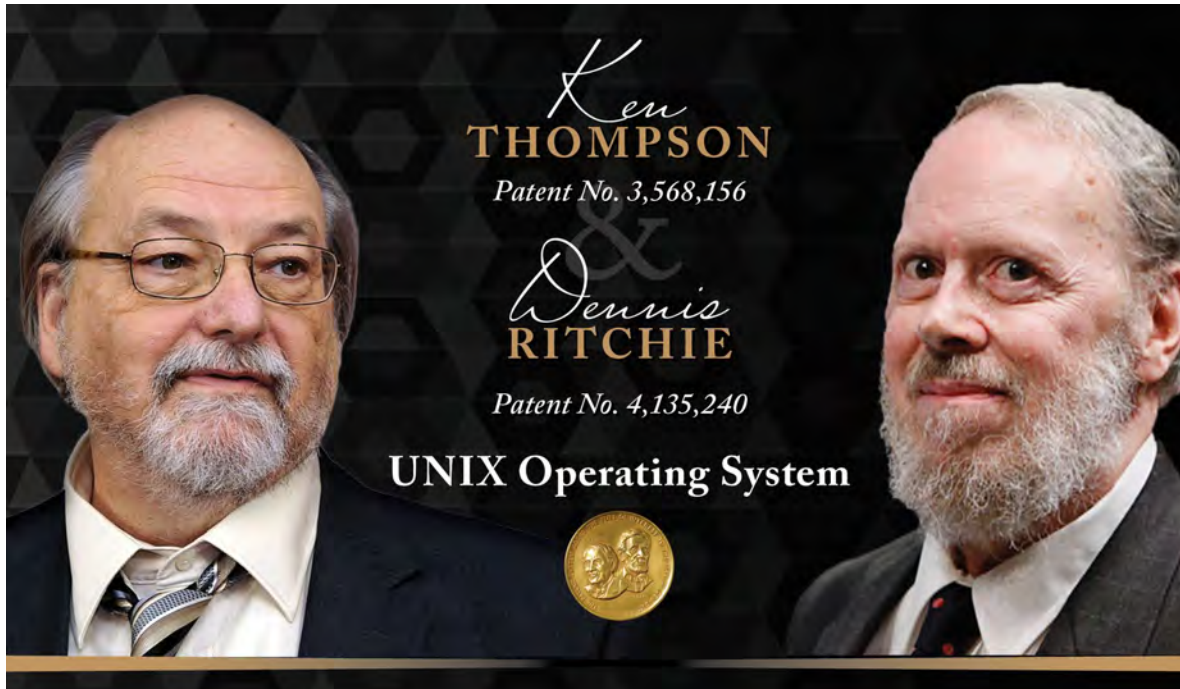






# The UNIX Operating System

- Developed in the 1970s after MULTICS proved too ambitious.
- UNIX is an operating system for programmers.
  - There are many versions of UNIX.
- More computers run a version of UNIX than any other operating system.
  - Most phones, routers, data centers, ... run some version of UNIX.
- Why has it lasted so long?
  - It has an effective programming model.
  - Simplicity and elegance do matter.



You must use UNIX  
(Linux/Ubuntu)

- You *do not* have a choice in this matter.
  - No, you must not use a Windows IDE.
  - If we catch you, then you get a zero.
- a. You can run Linux native on your computer.
- b. You can install VirtualBox and run Linux on that.
- c. You can do your programming on:  
[cse13s.soe.ucsc.edu](http://cse13s.soe.ucsc.edu)
- I personally favor the virtual machine approach.

You are fortunate to have access to many resources available on-line:

<https://proquest-safaribooksonline-com.oca.ucsc.edu>

This includes most of the O'Reilly books.  
*Be smart:* make good use of this resource.

Here are some of the books you will want to consult:

Loeliger & McCullough, *Version Control with Git*  
Newham, *Learning the bash Shell*  
Robbins, Lamb & Hannah, *Learning the vi and Vim Editors*  
Robbins, Siever, Figgins & Robert Love, *Linux in a Nutshell*  
Crawford & Prinz, *C in a Nutshell*  
Mecklenburg, *Managing Projects with GNU Make*



## O'Reilly Safari Books



July 12 - Fever still around 102°, cough not quite as bad, taking attacks.  
This evening I moved down to the room downstairs. After supper, Horace brought in 2 salmon (6 riyals). I inspected them well, and outside of slight drying, I think they were in good condition.

July 13 - Fever and continued coughing. Dr. Bonet visited this morning.

July 13 - Letter mailed to Lucille via Shekran. Dr. Bonet visited this morning. Dr. Bonet's fever has gone down 101.8°F. Still have danger of cough, and keep taking codine to stop it. Taking sulphur, atalaine, and codine for my cough. Last night I got sick after one hour in bed, and I just made it into the first floor, by the side of my teeth and it was very thin, and I lost all my coo-hiss. When I got back to bed a second I felt woe-h, but I slept very well during the night. Temp. of my fever 101.8° at 6:00 P.M.

(Cont.) July 13 - Dr. Bonet came this morning. He has been here today and so far I have kept it down. My owners are very loose and moved at least 7 times including one during lunch, right in the middle of my lunch. Dr. Bonet, came about 10 P.M. My fever is a little down 101.5 compared with last night. Dr. Bonet is doctor at the American Legation and his wife is a nurse. He is a Frenchman, and has worked in Indo-China. He has been here in Jidda, only several weeks. He is rather short and stoutish with a Jewish nose, and he wears eye glasses and has thinning black hair. He appears to be very competent in his profession. I made great decision to stick with the Museum, and do more for him.

July 14 - Dr. Bonet visited this morning. Temp. 101.5°F. I worked hard all day on specimens & notes, and I gave Horace all my fishing gear to use. He was very willing, and he may pick up more stuff before I leave Jidda. Temp. 101°F at 6 P.M. passionate love letter to his wife, given to Mr. Campbell and said to her. Dr. Bonet came again this afternoon.

Juan Domingo Suarez - Puerto-Rican fisherman, working east of Cape San Juan, P.R. There seem to be more abundant varieties in the Red Sea, than the Persian Gulf. This entirely due to the bottom character, of the Persian Gulf. No porpoises seen on the Red Sea in the vicinity of Jidda, also no

... vomiting after meals, but I didn't lose much and arrested the cough with codine. July 15 - Vital statistics of my fever and cough. pills total = 251,  $\text{CaSO}_4$  2, sulphur 48, 144  $\text{CaSO}_4$ , 80,  $\text{OH}_2$ .  
July 15 - Dr. Bonet still has my eraser. I have cash on hand as of today 100.28 guereche. The fish were collected from Zabab Ashra. The last night by Horace. I certainly appreciate his work, and I told him so. Thus I completed my half work of the day.

July 15 - Dr. Bonet came about 10 P.M. My fever is down to 100.5°F. I am feeling much better, and I find that codine is excellent for controlling deep coughing. It has saved vomiting on countless times the last week. Dr. Bonet visited about 6 P.M. July 16 - The I did not feel too hot I slept most of the day in brief periods. I had a midnight lunch. Am still living on codine.

July 16 - Dr. Bonet came about 10 P.M. My fever is down to 100.5°F. I am feeling much better, and I find that codine is excellent for controlling deep coughing. It has saved vomiting on countless times the last week. Dr. Bonet visited about 6 P.M. July 16 - The I did not feel too hot I slept most of the day in brief periods. I had a midnight lunch. Am still living on codine.

July 16 - Dr. Bonet came about 10 P.M. My fever is down to 100.5°F. I am feeling much better, and I find that codine is excellent for controlling deep coughing. It has saved vomiting on countless times the last week. Dr. Bonet visited about 6 P.M. July 16 - The I did not feel too hot I slept most of the day in brief periods. I had a midnight lunch. Am still living on codine.

July 16 - Dr. Bonet came about 10 P.M. My fever is down to 100.5°F. I am feeling much better, and I find that codine is excellent for controlling deep coughing. It has saved vomiting on countless times the last week. Dr. Bonet visited about 6 P.M. July 16 - The I did not feel too hot I slept most of the day in brief periods. I had a midnight lunch. Am still living on codine.

## Taking Notes

- Taking notes is essential for success in this course.
- Doing so means you can use them on tests.
- You cannot rely on simply reading the slides.
- A lot will be said that is not on the slides.
- You should turn in your notes before Monday of the following week using git.
- PDF or plain text, Word will not be accepted.
- If you do, you get a prize: you can use them to create your cheat sheet!
- Your goal in this course is to learn, not to pass.
- Unlike high school, you do not cram, test and forget.
- Many of you will try, and you will not succeed.



## Use of Electronics

- All electronics are forbidden during quizzes and examinations.
  - That means phones are *off* unless you have a waiver.
- You can use your computer or tablet to take notes.
- Your phone should be off, set to silent or vibrate during class.
- During class you may not watch videos, shop, play video games, or post on social media.
  - If you do, then you will be ejected from Zoom.





## Secrets of Success

- Start projects early.
  - You can always finish early and take the last day off ...
- Write your design document before writing code.
  - *Think* first, code second.
- Use debugging and analysis tools.
- You can study together but
  - All code must be your own!
- If you don't understand something, *then ask*.
  - Do not wait until the last minute!



# Programming Aphorisms

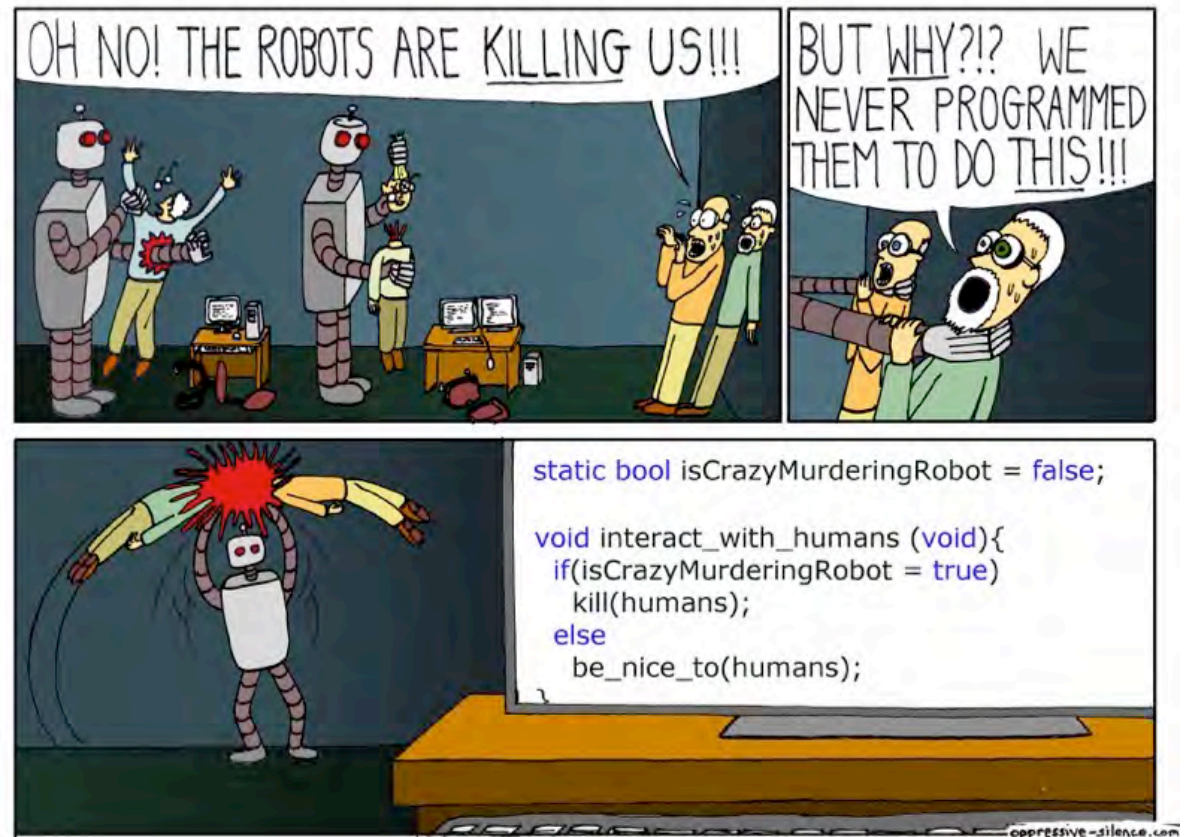
- Computers do exactly what they're told, not what you want them to do.
- Finding bugs can be time consuming.
  - When you fix one bug, you may introduce new ones.
- Attention to detail matters.
- Time spent *thinking* is repaid manifold in time saved coding.
- Take the time to draw diagrams, work through examples, ... it will pay off.
  - Coding is like poetry; you should go through multiple drafts.



R. Crumb, a true *Subgenius*

# Attention to Detail Matters

Do you see the bug in this code?





## Cheating

- You should all know what cheating is,
  - But in case you do not, you will *all* be signing an acknowledgement regarding it.
- Using code that you did not write is cheating,
  - Including code on the Internet.
- If you use code from others, including the TA, Grader, Tutor, ... you must acknowledge that person.
- You had better not use Chegg or similar web sites.

# Examples of Prohibited Activity

- You may not work on your assignment with anyone.
- You may not show your code or design to anyone.
- You may not have anyone “walk you through” an assignment, describe a solution in detail, or sit with you as you work on it.
  - This includes hired tutors.
- You may not post code from your project on-line in questions when you ask others for help.
- You may not hire *some guy from Pakistan* to write your code for you.
  - Yes, people have tried this (and failed).
- This list is *not exhaustive*.

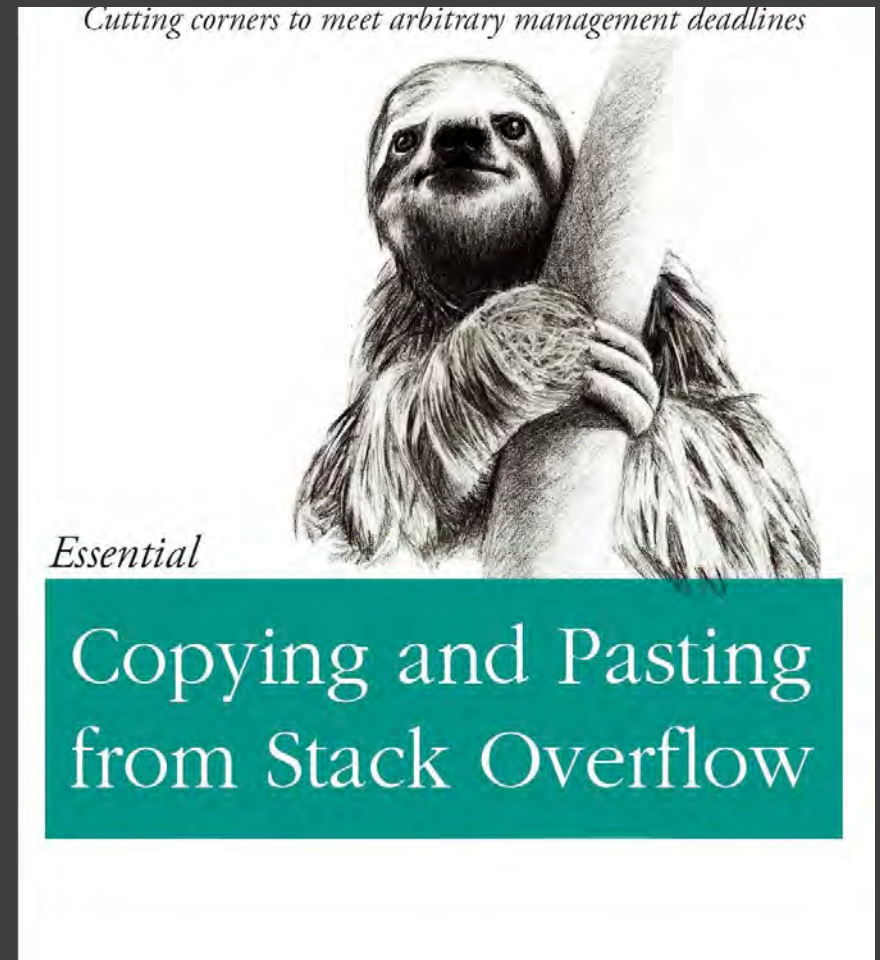


Klaus Fuchs  
Engaged in *very prohibited* activities



# The Internet does not have all of life's answers

- Avoid the temptation to “just Google it”
  - Your task is to *think* and to *learn*.
- We know how to use the Internet too, and if you copy code we will probably catch you.



# Programming Schedule

- Programs are assigned by *Saturday night*.
- Design documents are due *Thursday night*.
  - Preliminary design is 50% of the grade, and
  - Final design is the other 50% of the grade.
- Programs are due on *Sunday night*.
- No design on Thursday? *Zero* on the design document — no excuses!



# README.md

- Describes:
  - What your program does,
  - How to build it,
  - How to run it, and
  - Any issues that it has.
- Must be in *Markdown* format.



# DESIGN.pdf

- Describes how you will approach writing the program.
  - It should be at a level that a programmer of your level could write the program.
  - It should not include C code —
    - This is design, not programming.
    - Pictures, diagrams, pseudocode are what we want.
- It must be in PDF.
  - The first version must be pushed on Thursday.
    - This is not final but must show substantial effort.
    - This counts 50%.
  - The final version must be pushed on Sunday.
    - This counts for 50%.
  - If you do not push on Thursday, you get zero.



# Late Policy

- If you turn in your programs late, then you lose:
  - 25% each day for up to 3 days.
  - After 3 days you get a *zero*.
- There are very few valid excuses.
- “*But the dog ate my computer!*”
  - Using `git` lets you recover!
- You’ve been admonished to start early!



# Summary

- This class is *not* just another hoop for you to jump through:
  - You will use what you learn here from now on.
- Take notes.
- Ask questions.
- Read the book.
- Write the code.
- Do not cheat:
  - It hurts *you* most of all.
- Succeeding in this class will make subsequent classes *much* easier.
- Your *success* is our goal!

