(Text?) Editors

Learning Basic Math

```
% echo "#include <iostream>
> using namespace std;
>
> int main() {
>     cout << "2 + 2 = " << 4 << endl;
>
>     return 0;
> } " > math.cpp
```
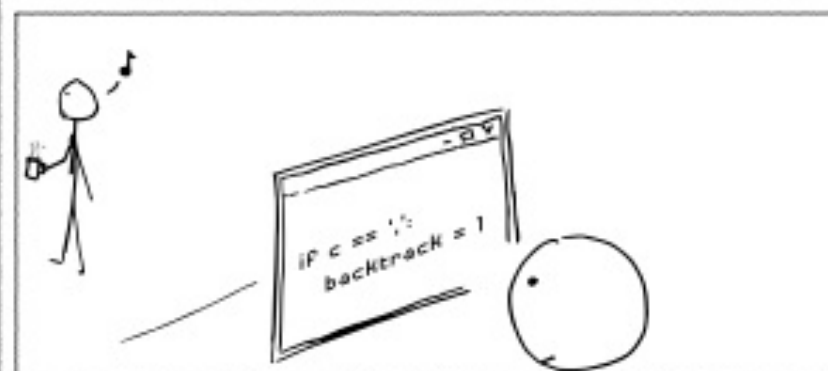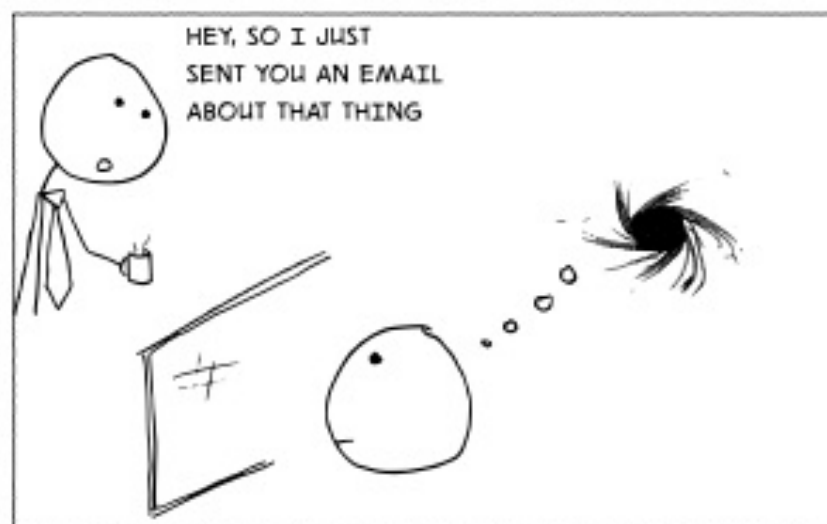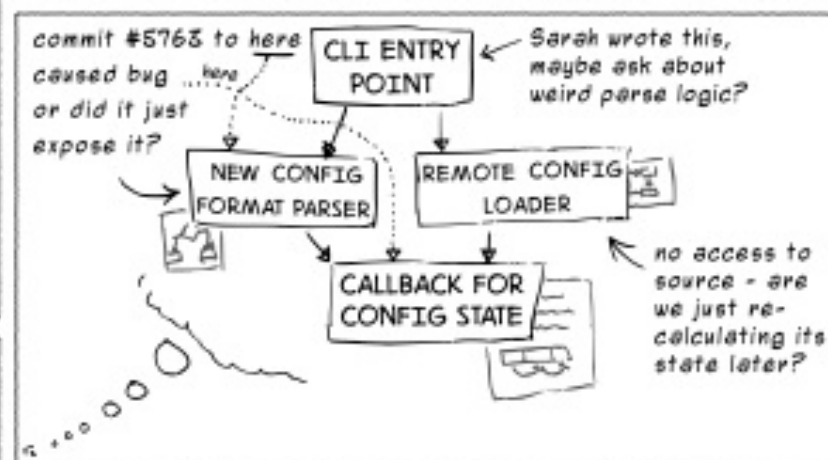
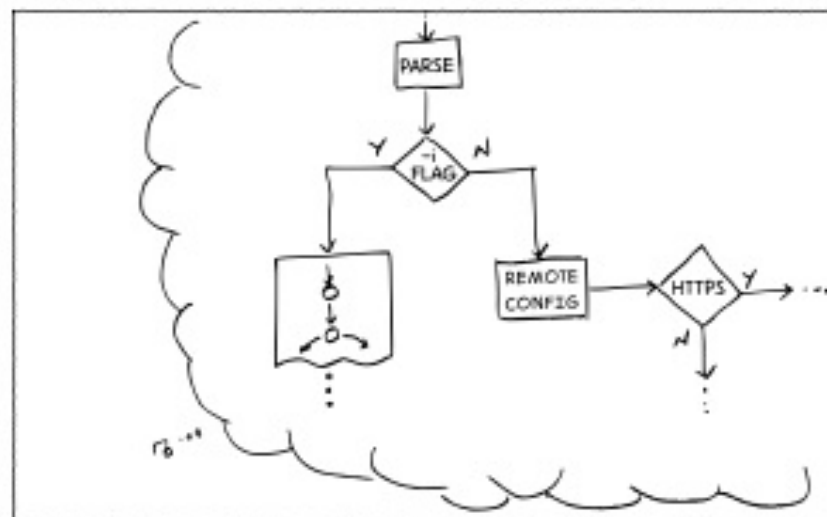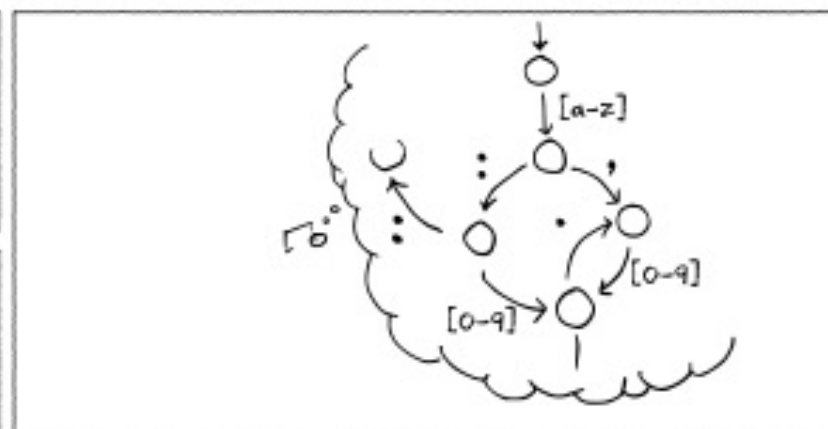- That was cheap!
- Let's check it out...

Quick Game: Name an editor

Quick Game: Name an editor

- https://en.wikipedia.org/wiki/Comparison_of_text_editors

THIS IS WHY YOU SHOULDN'T INTERRUPT A PROGRAMMER
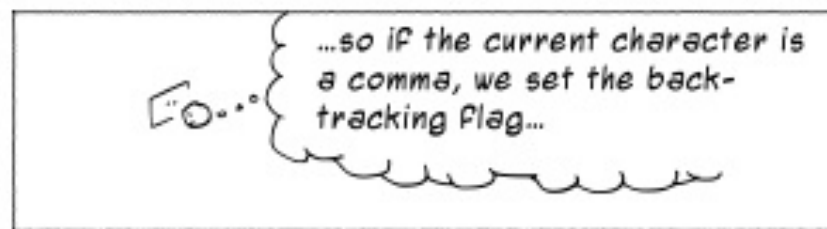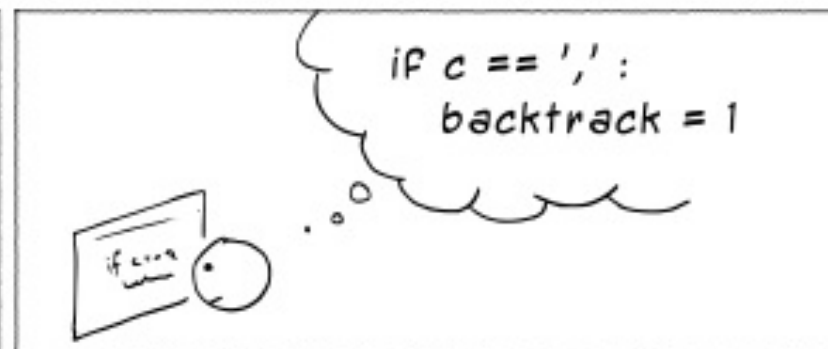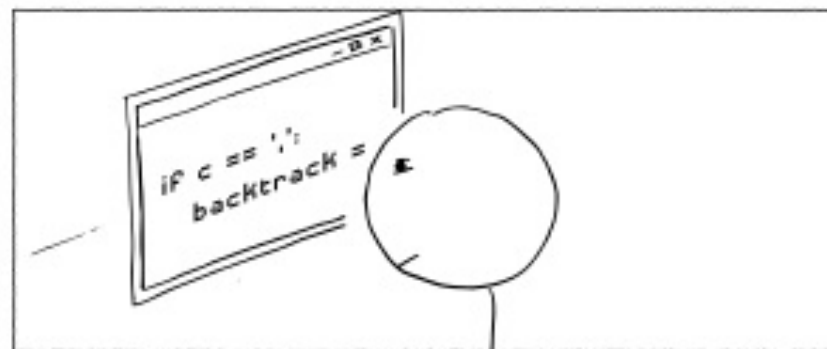
Editors are the stuff between your head and the screen

- You can walk and talk
- You can walk and read
- Could you walk and code?

Editors are the stuff between your head and the screen

- You can walk and talk
- You can walk and read
- Could you walk and code?

Writing code is like writing text, until it isn't

- Organize an outline
- Choose a section, author a cohesive narrative
    - How long of a paragraph can you hold in your head?
    - How many lines of code can you hold in your head?

Editors are the stuff between your head and the screen

- You can walk and talk
- You can walk and read
- Could you walk and code?

Writing code is like writing text, until it isn't

- Organize an outline
- Choose a section, author a cohesive narrative
  - How long of a paragraph can you hold in your head?
  - How many lines of code can you hold in your head?

Programmers are translating all the time

- Between what you want the machine to do (concept/spec) and *how* you tell the machine to do it (code)

Editor Wars: An Educated Guess

- Programmers fight about silly things
  - Tabs vs Spaces, Where to put Braces
- That may not be so silly

Editor Wars: An Educated Guess

- Programmers fight about silly things
  - Tabs vs Spaces, Where to put Braces
- That may not be so silly

It is all about what best minimizes the friction between what's in your head and what's on the screen

- People tell me DVORAK is faster than QWERTY[1], but probably not for me

Lecture Today

- **The Basics**

  1. Marcus struggles mightily with emacs

  2. And then does great in vim

  3. We cover where the mouse is a winner

  - *Goal:* Minimal competency in everything, so you can work with others

- **The Fancy Stuff**

  1. In "old" editors

  2. In "new" editors

  3. And what each other can't do

  - *Goal:* Exposure to the kind of things you can do. What is most useful is very personal.

Chart? Chart.

	emacs	vim	gedit
------------------------	------------------	----------------	-------------
Save file	`C-x C-s`	`:w`	click save
------------------------	------------------	----------------	-------------
Quit **without** saving	`C-x C-c n yes` `M-x kill-emacs`	`:q!`	click quit click no
------------------------	------------------	----------------	-------------
Save and quit	`C-x C-s` then `C-x C-c` `C-x C-c y`	`:w` then `:q` `:wq` `:x` `ZZ`	click save click quit
------------------------	------------------	----------------	-------------

Chart? Chart.

	emacs	vim	gedit
----------------------	---------------	--------------	------------
Save file	`C-x C-s`	`:w`	click save
----------------------	---------------	--------------	------------
Quit **without** saving	`C-x C-c n yes`	`:q!`	click quit
	`M-x kill-emacs`		click no
----------------------	---------------	--------------	------------
Save and quit	`C-x C-s` then `C-x C-c`	`:w` then `:q`	click save
	`C-x C-c y`	`:wq`	click quit
		`:x`	
		`ZZ`	
----------------------	---------------	--------------	------------

A digression: w vs x (vs ZZ)

- `:w` always writes, `:x` only if changed
  - Why do you care?
  - It's a bit annoying to need to care

Chart? Chart.

	emacs	**vim**	**gedit**
Save file	`C-x C-s`	`:w`	click save
Quit **without** saving	`C-x C-c n yes` `M-x kill-emacs`	`:q!`	click quit click no
Save and quit	`C-x C-c y`	`:x` or `ZZ` `:wq`	click save click quit
Enter some text	...just type	`i` then type	...just type
Search for a string	`C-s`	`/`	make search appear click search

"Enter some text" in vim, oh the many ways

`i` is the easiest, yet probably least used, way to enter text

- `i` - Go into insert mode here
- `I` - Go into insert mode at the beginning of the line
- `a` - Go into insert mode right after here
- `A` - Go into insert mode at the end of the line
- `o` - Insert a new line below and go into insert mode
- `O` - Insert a new line above and go into insert mode
- `s` - Delete this character and go into insert mode
- `S` - Delete this line and go into insert mode
- `C` - Delete the line from here and go into insert mode
- `R` - Enter *replace* mode

Reviewing some old magic

Remember Homework 2?

```
$ wget 'https://drive.google.com/uc?id=0B4qlH840ZwikbkZLS3Z5YTVSeW8&export=dow
nload' -O eecs280-w15-p2.tgz
$ tar -xf eecs280-w15-p2.tgz
$ echo -e '#include <cassert>\n#include "p2.h"\n' > p2.cpp
$ grep ';' p2.h | grep -v ' \*' >> p2.cpp

$ vim p2.cpp
# press 'jjj'   so that your cursor is on the i of int sum(...
# press 'qq'    to begin recording a macro into the vim register q
# press 'f;'    jump to the ';' character
# press 's'     remove character under the cursor and enter insert mode
# insert the needed text:
# '<space>{<enter>assert(false);<enter>}<enter><escape>'
# press 'j'     so that your cursor is on the i of int product(...
# press 'q'     to finish recording the macro
# press '16@q'  16 times play the commands stored in register q
```

"Vim sucks because"…

it can't copy / paste

- Yes it can, you just `y`ank and `p`aste instead
- And to cut you off, you can use `"+y` and `"+p` to yank and paste from the system clipboard

you can't use the mouse

- Yes you can, you just have to `set mouse=a`

the defaults are ~~terrible~~ not great

- Now you're getting somewhere

the time and pain required to learn it aren't worth the payoff

- I am honestly unsure.

This class in a nutshell:

- You are willing to type the same command over and over until you aren't
  - So you learn about the up arrow
- You master the game of up-up-up-enter up-up-up-enter until you drive yourself nuts
  - So you learn how to put these commands in a script
- *You might be willing to master a new skill faster if you knew it was out there, and someone guided you towards how to do it*

The goal is to expose you to many things, hope you master half, and can come back and pick up the rest when you're ready

- We do try to emphasize the ones worth mastering first
- Like version control. Seriously. Version control. All the things. Always.

A quick tour of some other editors and what they can do

- "General purpose" editors
  - Atom
  - Sublime
- Integrated development environments
  - XCode
  - Eclipse
  - Visual Studio