nano? REAL PROGRAMMERS USE emacs

HEY. REAL PROGRAMMERS USE vim.

WELL, REAL PROGRAMMERS USE ed.

NO, REAL PROGRAMMERS USE cat.

REAL PROGRAMMERS USE A MAGNETIZED NEEDLE AND A STEADY HAND.

EXCUSE ME, BUT REAL PROGRAMMERS USE BUTTERFLIES.

THEY OPEN THEIR HANDS AND LET THE DELICATE WINGS FLAP ONCE.

THE DISTURBANCE RIPPLES OUTWARD, CHANGING THE FLOW OF THE EDGY CURRENTS IN THE UPPER ATMOSPHERE.

WHICH ACT AS LENSES THAT DEFLECT INCOMING COSMIC RAYS, FOCUSING THEM TO STRIKE THE DRIVE PLATTER AND FLIP THE DESIRED BIT.

NICE. 'COURSE, THERE'S AN EMACS COMMAND TO DO THAT.

'C X M - C M - BUTTERFLY...

DAMMIT, EMACS.
• Introductions
  – Name
  – Where from
  – Something non-academic

• What's the point of this class?
  – Not history, but not all cutting edge
  – Awesome! We get to learn a lot!
  – Bummer – we have to learn a lot.

• Class webpage

• Syllabus
  – use of moodle
  – email
• Labs
  – When due
  – Partners
  – How independent
  – Learning C
  – vs. Projects – may have more than in past

• Reading
  – Our textbook
  – Before/after lecture
  – Questions at beginning of class

• Active Learning
• C
  – Why is C special / different / awesome / horrible?

• How does the hello world code turn into a program?

• Source program (human) -> preprocessor
• modified source program (human) -> compiler
• assembly program (assembly) -> assembler
• relocatable object programs (binary) -> linker
• executable object program (binary)
Pre-processor (cpp) → hello.i → Compiler (cc1) → hello.s → Assembler (as) → hello.o → Linker (ld) → hello
g++ helloWorld.c -o helloWorld
./helloWorld
• How does the hello world program execute?
• disk ➔ memory ➔ register
• display device
• Memory hierarchy

• Virtual addresses, heap/stack, files
Linux

What is it?
How it fits into the class....
Woo!

- Done with chapter 1!
- Only about 8 chapters to go!
Overflow

• Number
Overflow

- Number
- Buffer
  - Smashing the stack for fun and profit
  - Phrak 49
  - Generally from strcpy(), strcat(), and sprintf()
  - Overflow fixed size buffer with machine code
Bits

**Electronic Implementation**

- Easy to store with bistable elements
- Reliably transmitted on noisy and inaccurate wires

![Diagram showing voltage levels and bit representation](image)
Bits

- 101
- 5 in binary
- 5 if it’s an integer
- $+\infty$ if float
- 0x101 = 65 in dec
- A if it’s hex referring to ASCII
Bits + Context

• Integer
• Float
• Character
• Program instruction
• Memory Address

• Bytes
  – Trivia: Nibble/Nybble