COS 109 basic info

- Brian Kernighan
  bwk@cs.princeton.edu   www.cs.princeton.edu/~bwk
  office hours: right after class or make an appointment

- Archie McKenzie, archiem@princeton.edu, Wed & Sun 6-9

- web site: www.cs.princeton.edu/courses/archive/fall22/cos109
  (we won't use Canvas very much)

- please fill out the survey (link is also on web site)
  https://forms.gle/cZYdW7xSdPMi3Rfp6

- first problem set is due midnight Wednesday Sep 14
- first lab is due midnight Sunday Sep 18
  (both are posted on course web page)
House rules

• turn off your phone and laptop
  – it helps to keep you and me engaged

• let me know if there's anything I can do to make this work better

• COVID precautions
  – for now, please be vaccinated, boosted, masked, socially distanced

• ask questions / make comments / … about anything any time

• questions so far?
Administrivia  (check the web page for updates!)

• notes will be posted online
  – but not everything will be in them or in the textbook
• readings: ~ 1 hour/week, before class
  – textbook readings are important; most others are cultural enrichment
• 8 problem sets: ~ 1-2 hours each
  – posted by Wednesday, due following Wednesday by midnight
• 8 labs: ~ 2-3 hours each, plus reading to prepare
  – posted by Sunday, due following Sunday by midnight
  – do the labs on your own, any time
• open-book take-home midterm during midterm week
• open-book take-home final exam during December exam period
• grading (approximately):
  20% problem sets + 20% labs + 20% midterm + 35% final + 5% participation
• regular attendance at lectures is required; participation helps
Textbook

• 2nd edition is definitely preferable
  – get the paperback version!

• 1st edition is ok

• good supplementary reading if you're interested in privacy and security =>
Course outline

• hardware (3-4 weeks)
  – how computers represent and process information
  – what's inside a computer, how it works, how it's built

• software (3-4 weeks)
  – how we tell computers how to do things
  – a very gentle introduction to programming in Python

• communications + data (3-4 weeks)
  – how the Internet and the Web work
  – machine learning, artificial intelligence
  – threats and defenses: privacy, security, cryptography

• along the way
  – current events, history, QR / QCR, ...
Hardware: tangible devices and gadgets

• how computers represent and process information
  – universal digital representation of information:
    everything is represented as numbers
  – bits, bytes, binary

• a computer is a universal digital processor
  – it stores data and instructions in the same memory
  – the instructions are numbers
  – it's a general purpose machine:
    change the numbers and it does something different
  – your phone is a computer

• hardware has been getting exponentially smaller, cheaper, faster for ~60 years
Software: telling computers what to do

• algorithms
  – precise sequences of steps to perform various tasks
  – what's possible, what's feasible, what's efficient
    some problems are intrinsically very hard (we think)

• programs and programming
  – implementation of algorithms to be run on a computer
  – programming languages: how to express the steps
  – real programs: operating systems and applications

• software intellectual property issues
  – patents, copyrights, standards, ...
Communications: computers talking to each other

• the Internet is a universal digital network
  – depends on protocols, standards, agreements, cooperation

• we can easily communicate with people anywhere
  – we are visible to and accessible by strangers everywhere

• information passes through many sites
  – where it can be inspected, modified, blocked, slowed down, …

• personal privacy and security are at risk
  – tracking, data aggregation, surveillance (government and commercial)
  – phishing, identity theft, …
  – viruses, worms, bots, hijacking, trolls, disinformation, …

• everything on the Internet is vulnerable
  – cyber attacks
  – Internet of Things
It's not just computers

- computers and networking are spreading into devices
- devices are increasingly powerful
- devices and systems are increasingly connected to the Internet: "Internet of Things"

phones
games, toys
consumer electronics: Alexa et al, smart TVs, Fitbit, Ring, Nest, ...
cars (self-driving or not)
planes
medical systems and devices
infrastructure: phones, power, transportation, manufacturing, ...
weapons
...

...
Privacy

- data about shopping, banking, location, taxes, ..., is all digital
  - public records are increasingly digital too
    e.g., election contributions often include home addresses

- data is easy to collect, store, copy, analyze, sell

- technically, it's impossible to control access
  - we're vulnerable to bugs, incompetence, stupidity, theft

- legally, in USA, we don't control data about ourselves
  - anyone can collect and sell anything about all of us (and they do)
  - laws are different in different countries (e.g., European Union GDPR)
Security

- the universal network makes us vulnerable to strangers
  - the Internet has no geography
  - it's easy to lie about who you are and where you are
  - the bad guys are usually far away

- general-purpose computers are everywhere
  - web pages and email can contain programs
  - phone apps often contain spyware and malware

- leads to spam, phishing, viruses, spyware, ransomware, ...
  - tracking and surveillance by governments and businesses
  - theft by criminals everywhere

- it's impossible to control such programs
  - and to eliminate tracking and surveillance, trolling, fake news, influencing
Goals of the course

• understanding how digital systems work
  – hardware, software, communications
  – representation, processing, storage, transmission of information
  – principles, not just today's details and buzzwords
  – a handful of useful skills

• some sense of the past and possible futures
  – history, trends, potential, intrinsic limitations, tradeoffs

• some appreciation of computer science as a discipline
  – great ideas, algorithms, capabilities and limits of computers
  – and its usefulness in other academic fields

• useful QCR
  – numeracy: reasoning, estimation, assessing numbers, ...
  – judgment: do the numbers make sense? are they plausible?
  – enough programming that it's not a mystery

• intelligent skepticism about technology