CS-210, Technical History of Computing

1. Introduction

CS 210. Technical History of Computing. 3 credits, 3 contact hours (3;0;0).

Prerequisites: (CS 100 or CS 101 or CS 103 or CS 104 or CS 113 or CS 115 or BNFO 135) and any History and Humanities GER 200 level course and ENGL 101. This course is for students in computing majors. Students will gain a comprehensive overview of the evolution of computing from the start of recorded history through modern times. By studying history, you will understand the context of modern developments in CS/IT, including cyclical trends and why various approaches did or did not work. Learning where it all came from will also help young computer scientists to speak intelligently with older colleagues and managers in the workforce. Topics include mechanical calculating, analog computing, relay/tube computers, transistors, integrated circuits, I/O such as punch cards/paper tape/floppy disks, the minicomputer generation, the microcomputer revolution, development of graphical and network systems, early mobile computer, and modern history. A special focus on historic developments in New Jersey will be part of all lectures.

The course is led by Adjunct Instructor, Evan Koblentz. Click here to learn about me.

2. Five simple rules

There are five simple rules in my class. In no particular order:

- Pay attention. It is insulting to me and other students if you do homework for other classes, play games, chat online, etc. while you're here. If I hear typing, then I expect you're taking notes.
- Speak up! I am happy when students ask smart questions.
- Be civil. I encourage active debate about historical topics and how they impact current events. You may vehemently disagree with me or with other students, but keep it polite.
- Don't use the F-word. No, not THAT one. I am talking about "first". This is a history course, so we'll learn about generations of computing. "First" is subjective and we will aim to avoid it at all costs.
- Never cheat. It's unfair to yourself and your classmates. Cheating includes looking up answers online, using AI such as ChatGPT to write answers, sharing answers with other students, posting quizzes/exams/answers online, and so on. We will abide by the NJIT policy, which states:

“Academic Integrity is the cornerstone of higher education and is central to the ideals of this course and the university. Cheating is strictly prohibited and devalues the degree that you are working on. As a member of the NJIT community, it is your responsibility to protect your educational investment by knowing and following the academic code of integrity policy that is found at: http://www5.njit.edu/policies/sites/policies/files/academic-integrity-code.pdf. Please note that it is my professional obligation and responsibility to report any academic misconduct to the Dean of Students Office. Any student found in violation of the code by cheating, plagiarizing or using any online software inappropriately will result in disciplinary action. This may include a failing grade of F, and/or suspension or dismissal from the university. If you have any questions about the code of Academic Integrity, please contact the Dean of Students Office at dos@njit.edu.”

3. Logistics

The best way to reach me is evank@njit.edu or through the Canvas messaging system. Adjuncts do not have standard office hours. However I am a full-time NJIT staff member, so I am available most days by appointment (Monday-Tuesday: remote / Wednesday-Thursday-Friday: in-person). My office location is Fenster 440B. Please contact me first, rather than dropping in.

Grading is as follows: 30% final exam, 30% midterm, 20% weekly quizzes, 20% attendance.

The required book is "A new history of modern computing" (Haigh/Ceruzzi).

4. Schedule

- Class 1: Intro + Pre-history: What's a computer? Clay tablets, invention of number systems, analog vs. digital, abacus, quipu, Greek/Roman mechanical programming, Antikythera device, slide rule, binary, Pascal, Leibniz, Babbage/Lovelace, etc.
- Class 2: Giant brains: Boole, differential analyzers, Bush, relay computers (Bell Labs/Harvard), vacuum tubes, Turing, Colossus, ENIAC, National Bureau of Standards
- Class 3: Commercial hardware: punch cards/IBM, UNIVAC, LEO, von Neumann, group project (develop a computer startup company, circa 1957)
- Class 4: Commercial programming: the women of ENIAC, Shannon, UNIVAC programming, COBOL (with hands-on demo), Fortran
- Class 5: Transistors and integrated circuits: Edison effect (here in Newark!), Bell Labs, what else did transistors enable, MIT TX-0, MOBIDIC, Fieldata, Noyce/Kilby, Moore's Law
- Class 6: Minicomputing/analog: DEC PDP/VAX, Boston 128 corridor, analog's role, C, Unix, timesharing, MIT vs. NJ schools of thought, Mauchly again, EAI, future role of analog computing in quantum
- Class 7: Microcomputer revolution: RESISTORS, Amateur Computer Society, BASIC, People's Computer Club, Ted Nelson, Homebrew Computer Club vs. ACG-NJ, Mark-8, S-100 generation, Microsoft, Apple
- Class 8: Midterm Exam
- Class 9: Mainstream microcomputing: The trinity: Apple II, Commodore PET 2001, Radio Shack TRS-80 Model 1, BASIC group project (can you make a program from nothing but some examples and reference books?)
- Class 10: Networking: Hardware vs. applications — SAGE, ARPAnet/IMPs, BBN, NSFnet, Internet, Intel/DEC/Xerox make Ethernet; email, FTP, Archie, Veronica, Gopher, WAIS, WWW, modern 'apps'
- Class 11: Becoming graphical: Bush's As We May Think, SRI's Mother of All Demos, Xerox PARC, Mac, The Humane Interface, Windows + underdogs (Amiga, etc.)
- Class 12: Mobile: Curta, DYSEAC, MOBIDIC revisited, science fiction, Osborne, Compaq, laptops, Newton, Simon, etc.
- Class 13: Modern history: EIES, early websites, cyclical (cloud vs. timesharing), NVRAM, artificial intelligence, etc. / Vintage computing hobby
- Class 14: Review
- Class 15: Final exam