

Some History of Computers

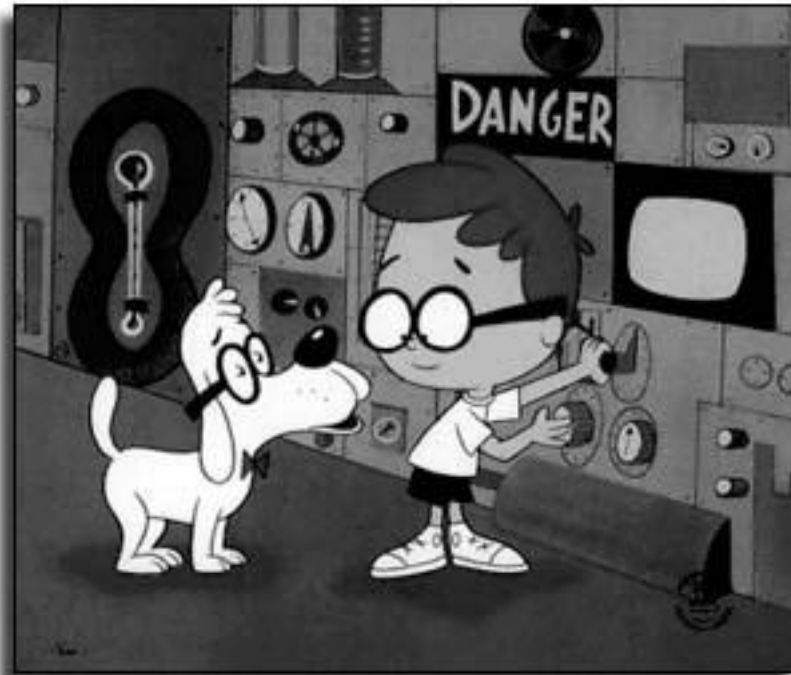
Provided by:

Wayne Heym (w.hey@ieee.org)

<http://www.cse.ohio-state.edu/~hey>

Copyright © 1998-2005 by Rick Parent, Todd Whittaker, Bettina Bair, Pete Ware, Wayne Heym

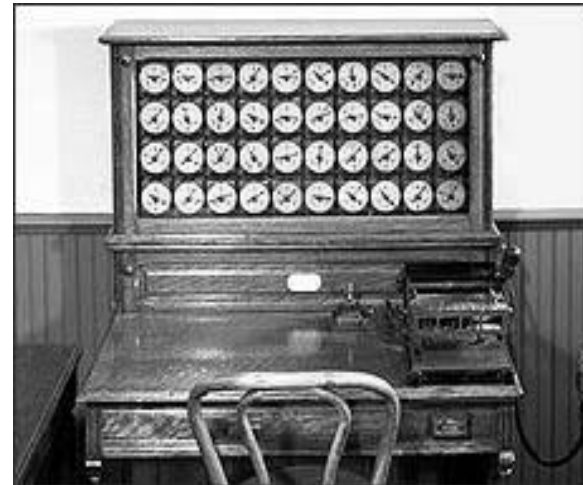
100+ Years of Amazing Computers



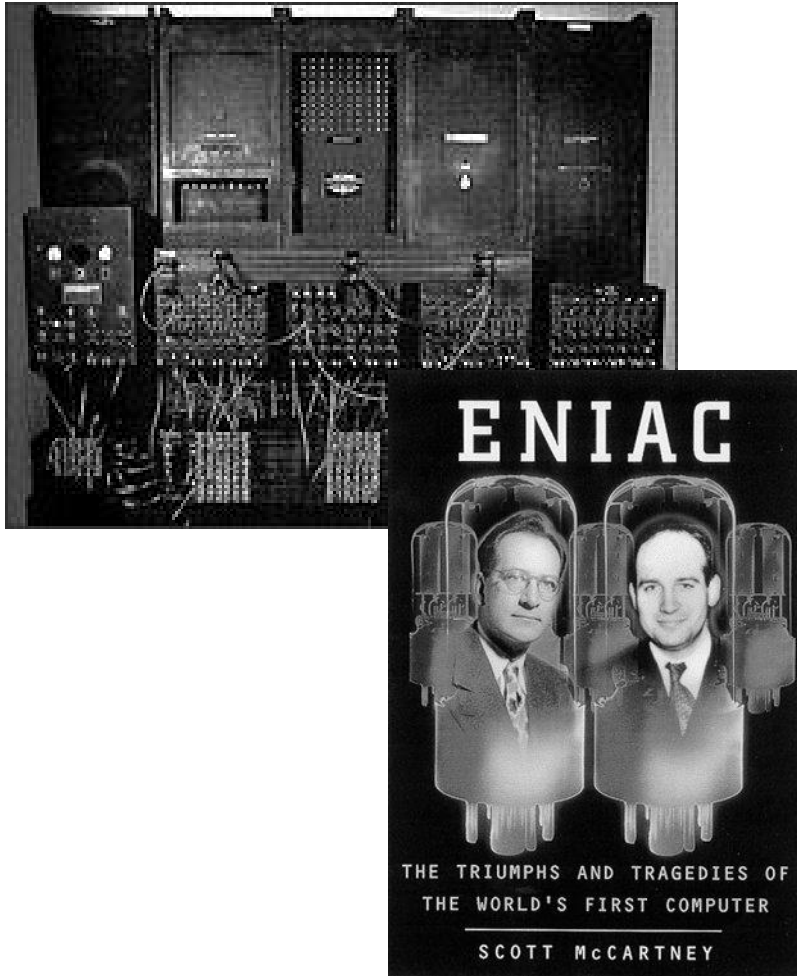
*Sherman, set the WayBack Machine to the year
1890...*

1890: Hollerith Tabulating System

- ◆ Census Counter
- ◆ Hollerith Tabulating System Was A System Of Machines
 - Punch,
 - Tabulator
 - Sorting Box
- ◆ Hollerith's Business Joined A Firm That Later Became IBM.

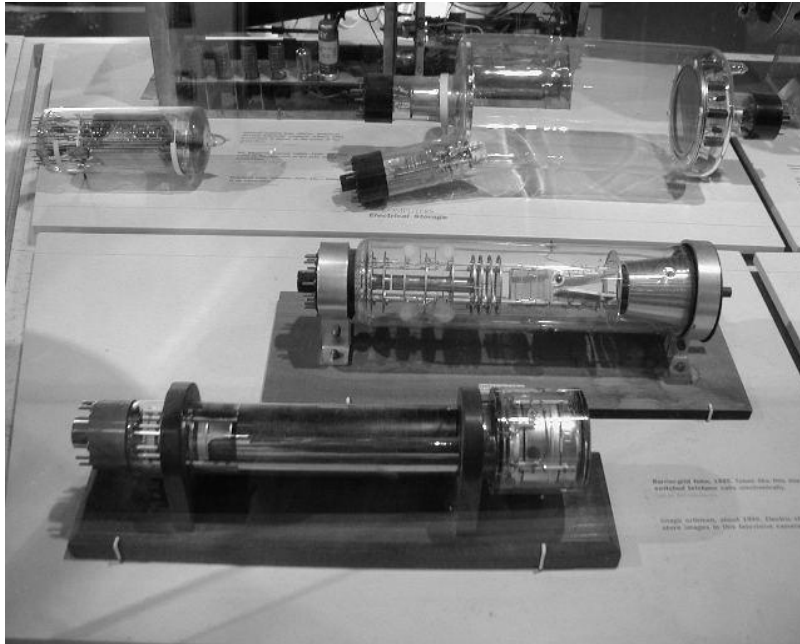


1943-45: *Eniac*



- ◆ Electrical Numerical Integrator And Computer
- ◆ Built To Compute Ballistics Tables For U.S. Army Artillery During World War II.
 - 1,000 Times Faster Than Any Existing Device.
- ◆ External Plug Wires Used To Program The Machine
- ◆ Principal Designers, J. Presper Eckert And John Mauchley
- ◆ Cost, About \$400,000

Vacuum Tubes

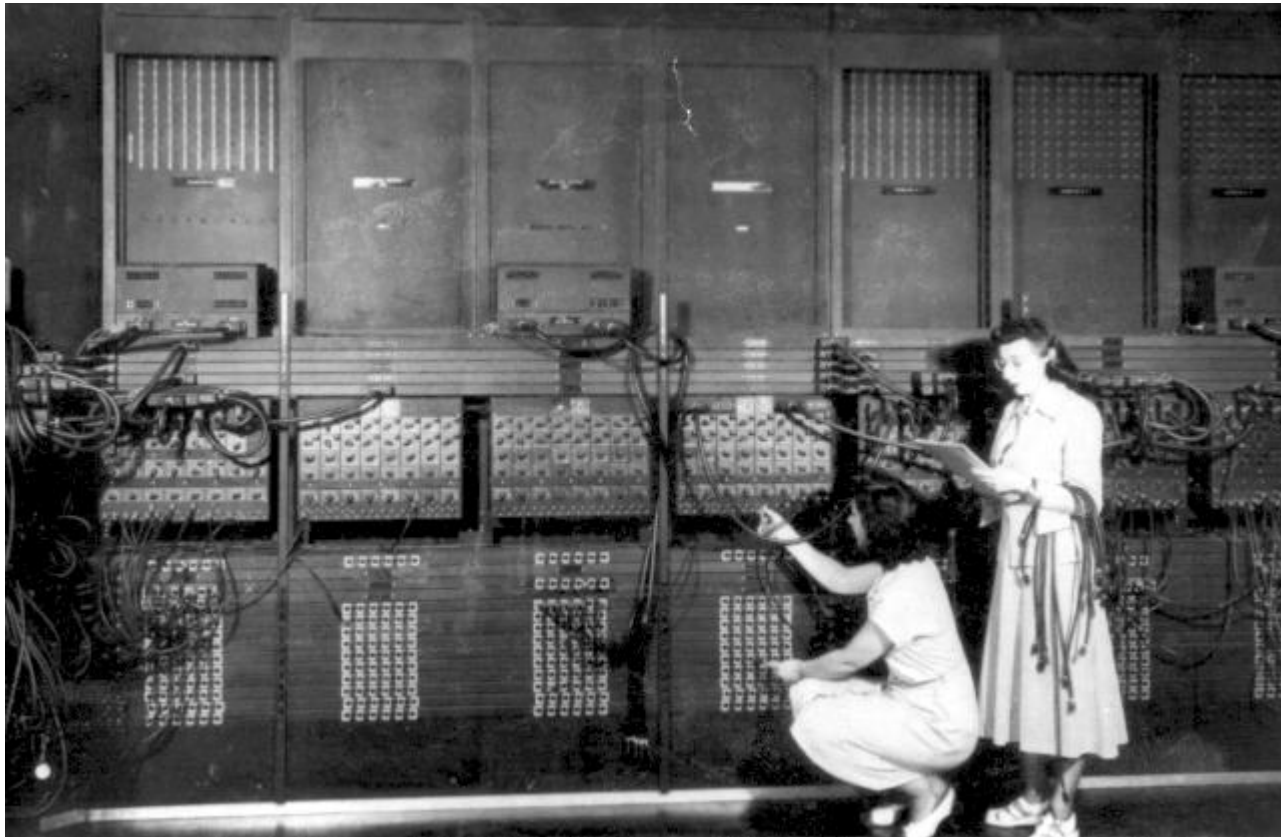


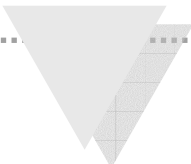
◆ ENIAC

- Used Some 18,000 Vacuum Tubes.
- 30 Feet By 50 Feet
- Weighed 30 Tons

The ENIAC was a decimal machine!

Programming the Eniac





Original Eniac Programmers



Kay Antonelli



Jean Bartik



Betty Holberton



Marlyn Meltzer



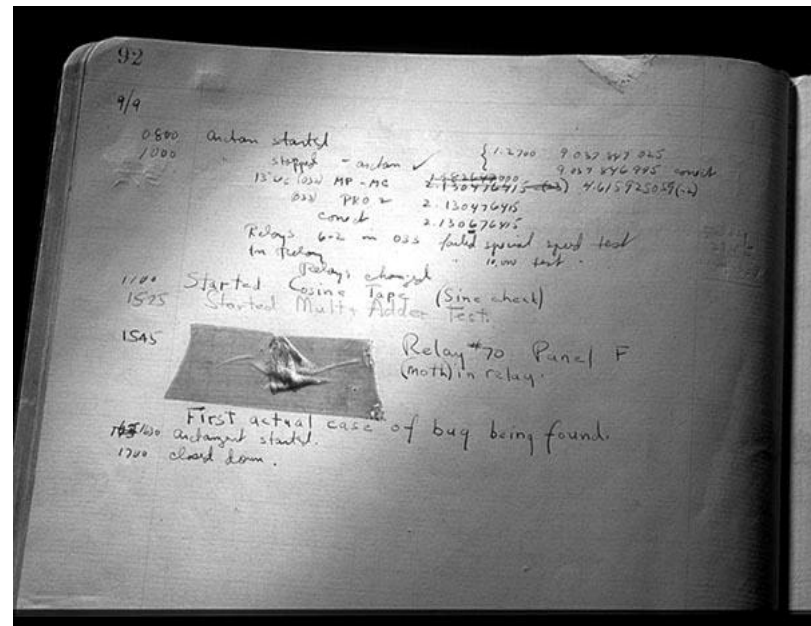
Frances Spence



Ruth Teitelbaum

The Bug

- ◆ In 1947, engineers found A moth stuck in one of the components.
- ◆ Taped it in their logbook
- ◆ Labeled it "first actual case of bug being found."



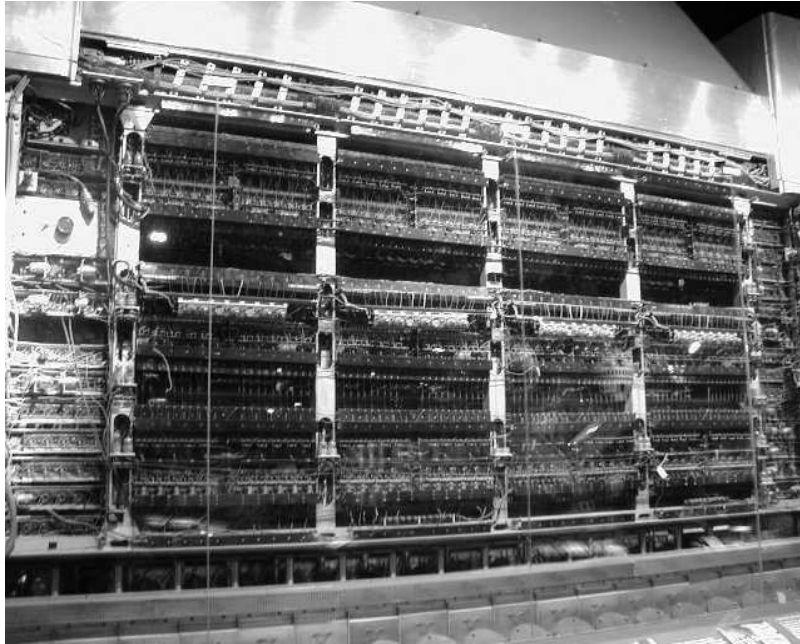
Grace Hopper (1906-1992)



Cmdr. Aiken and Lt. Hopper with parts of the difference engine, from *Christian Science Monitor*, March 20, 1946.

- ◆ 1953: Invented The Compiler
 - Translates English Language Instructions Into Language Of The Target Computer
 - "Lazy" And Hoped That "The Programmer May Return To Being A Mathematician."
- ◆ Led To The Development Of The Business Language Cobol.
- ◆ Retired From The U.S. Navy As A Rear Admiral.

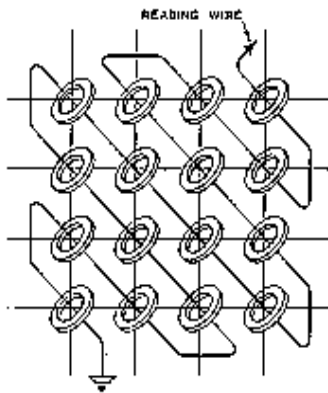
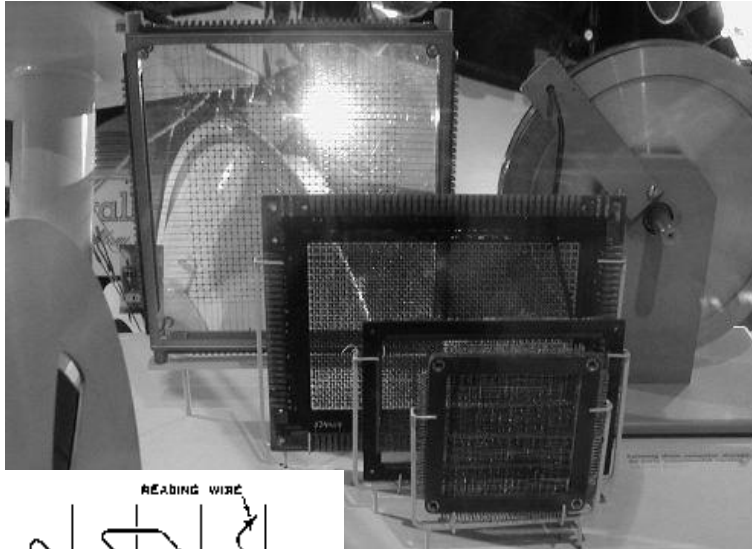
IAS (1946-1952)



- ◆ Institute For Advanced Study At Princeton University.
- ◆ Designed And Directed By John Von Neumann.
- ◆ Cost: Several Hundred Thousand Dollars.

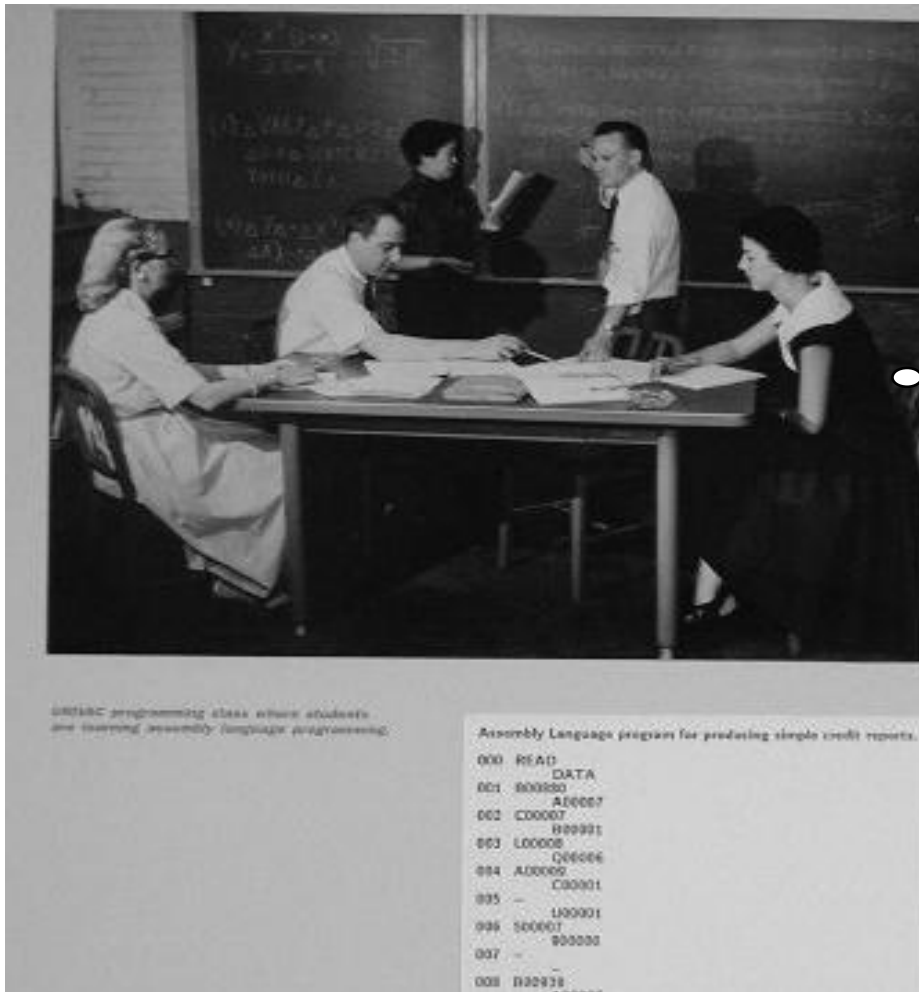
Used externally stored programs that could be loaded and executed.

1949: Core Memory



- ◆ A Small Ring, Or Core, Of Ferrite (A Ferromagnetic Ceramic) Can Be Magnetized In Either Of Two Opposite Directions.
- ◆ A Core Can Be Used For Storing One Bit Of Information.
- ◆ For Almost 15 Years, 'Core' Was The Most Important Memory Device.
- ◆ The Invention Of Core Memory In Was A Leap Forward In Cost-effectiveness And Reliability.

1950s Assembler Programming Class



*This would be
so much easier
with a
computer...*

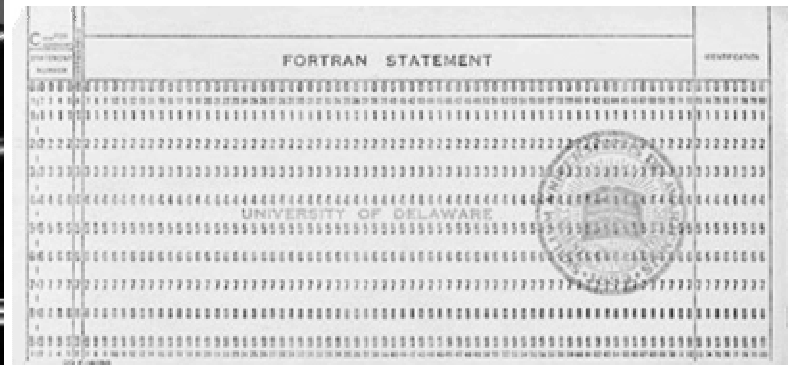
1965: PDP8



- ◆ Programmed Data Processor
- ◆ 50,000+ Sold
- ◆ Cost: \$18,000.
- ◆ Speed: 1.5 Micro-second Cycle Time
- ◆ Primary Memory: 4K
 - 12-bit Word Core Memory
- ◆ Power: 780 Watts

What does cycle time mean?

1960s/70s Card Reader



Card is pre-printed with FORTRAN field layouts

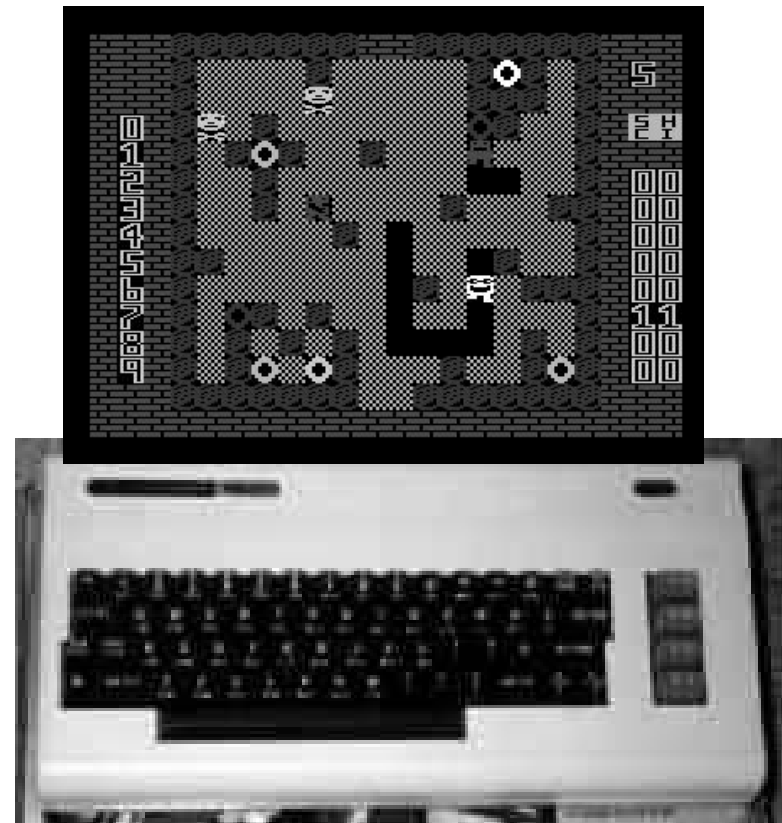
1977: Trs-80



- ◆ Radio Shack "Trash-80,"
- ◆ 4K Of Memory
- ◆ Could Not Handle Lowercase Letters
- ◆ Only Three Error Messages:
 - "HOW?"
 - ▼ Whenever The User Tried To Perform An Illegal Function
 - "What"
 - ▼ When A Syntax Error Occurred
 - "Sorry"
 - ▼ When The Available Memory Ran Out
- ◆ Cost Only \$400!
- ◆ Some 55,000 Machines Sold In First Year

1979: Vic-20

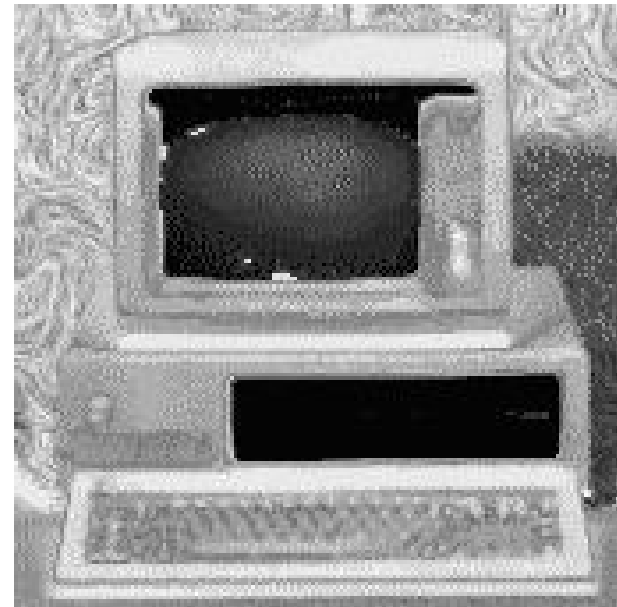
- ◆ Processor Speed: 1.0227 Mhz.
- ◆ ROM: 16kb
- ◆ RAM: 5kb (3.5kb User Memory)
 - Expandable To 32kb.
- ◆ Screen: 22 Columns By 23 Rows.
 - Character Dot Matrix: 8 By 8 Or 8 By 16 (User Programmable).
 - Screen Dot Matrix: 176 By 184 With Up To 16 Colors.
- ◆ Sound: 3 Voices Plus White Noise.
- ◆ Media: Tape Drive



Bettina's first PC!

1981: IBM 5150

- ◆ Processor: 8088.
- ◆ RAM: 16KB, expandable to 64KB
- ◆ 12" display
- ◆ Cassette drive and optional floppies
- ◆ 6 slots for plug-in cards
- ◆ Most importantly: I-B-M
- ◆ IBM hires pair to create operating system for PC, allowing the two to keep the marketing rights to DOS (who?)



1984: Macintosh



- ◆ Revolutionary Graphical User Interface (GUI).
 - A Device Called A Mouse
 - Pictorial Symbols (Icons) On The Screen.
 - Select Commands, Call Up Files, Start Programs, Etc.
- ◆ Original Selling Price: \$2,495

*What if you had to build your own computer
– from scratch?*





Course Objectives

- ◆ Understanding the architecture (how the computer executes assembly language instructions) is the more important aspect of a course at this level.
- ◆ The fundamental concept to understand is that everything in the computer is represented by ones and zeros (by electric current flowing or not flowing at a specific place, or by something being magnetized one direction or the other, etc.).



Course Objectives

- ◆ At the lowest level, this course will cover various binary formats of assembly language instructions and various ways in which data can be represented using ones and zeros and how these can be organized into a program.
- ◆ At high levels, assembly language programming techniques will be studied and a specific assembly language will be used to illustrate these techniques.



Homework #0-0

This homework is designed to help you to get acquainted with this course, and to get started.

- ◆ Surf to my webpage and find the following:
 - Office Location
 - Office Hours
 - Email Address
 - Course Overview
 - Course Schedule
 - Date/Time of the midterm and final examination



Homework #0-1

- ◆ Purchase the textbook written by Maccabe.
- ◆ Find the reading assignment for the second class-meeting in the CSE 360 Course Schedule, and read the assigned material.
- ◆ Pledge to do the reading assignment before each class meeting.



Homework #0-10

- ◆ Login to your CS unix account using the user name provided by your instructor.
- ◆ The domain name of the machine you login to is stdsun.cse.ohio-state.edu.
- ◆ Your initial password is the last four digits of your social security number followed by your first and last initials.
 - For example, Luke Skywalker, whose social security number is 123-45-6789, has a password of 6789ls.



Homework #0-11

- ◆ Find the class newsgroup
 - `news://news.cse.ohio-state.edu/cse.course.cse360`
- ◆ Find and read the story of **Mel, A Real Programmer, and Real Programmers Don't Write Specs**
 - *Are you a real programmer? Do you want to be?*
- ◆ Pledge to read the class newsgroup each day.



Homework #0-100

- ◆ Look over the Course Packet
 - It contains a wealth of information crucial to pursuing this course of study, including in-class exercises, and the overhead transparency slides.
- ◆ Plan to bring the Course Packet to each class meeting after today.



Homework #0-101

- ◆ Find answers to the following questions.
 - If you fail to electronically submit a homework or laboratory assignment by the due date/time
 - ▼ Will you later be able to receive any credit for that assignment?
 - Suppose you have electronically submitted on time, but, later, you electronically submit again, but after the due date/time.
 - ▼ Will you receive any credit for that assignment?
 - Assignments are acceptable only as plain text files; be sure you know what a plain text file is.
 - Where can you find Homework #1 and later assignments, as they become available?



Homework #110

- ◆ In the Course Packet, read the section of the Syllabus on “Academic Misconduct.”



Make a Table on an Index Card

- ◆ Show Different Representations of Numeric Values.
 - Column Headings Should be:

Decimal	Octal	Hexadecimal	Binary



One Row for Each Numeric Value.

- ◆ Show, in Increasing Order,
 - Representations for 0, 1, 2, 3, 4, ... 20
 - Then, 2^5 , 2^6 , ... 2^{16}
 - Finally 2^{20} , 2^{30} , 2^{31} , 2^{32}

For Example,

<i>Decimal</i>	<i>Octal</i>	<i>Hex</i>	<i>Binary</i>	<i>Note</i>	<i>Roman</i>	<i>Nat'l Lang</i>
0	0	0	0			zero
1	1	1	1	2^0	I	one
2	2	2	10	2^1	II	two
And so on.						
20	24	14	10100		XXIV	Twenty
32	40	20	100000	2^5	XXXII	..
And so on.				..		
				2^{16}		
				2^{20}		
				2^{30}		
				2^{31}		
				2^{32}		



Information Representation 1

- ◆ Positional Number Systems: position of character in string indicates a power of the base (radix). Common bases: 2, 8, 10, 16. (What base are we using to express the names of these bases?)
 - Base ten (decimal): digits 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 form the alphabet of the decimal system.
 - ▼ E.g., $316_{10} =$
 - Base eight (octal): digits 0, 1, 2, 3, 4, 5, 6, 7 form the alphabet.
 - ▼ E.g., $474_8 =$



Information Representation 2

- Base 16 (hexadecimal): digits 0-9 and A-F.
 - ▼ E.g., $13C_{16} =$
- Base 2 (binary): digits (called “*bits*”) 0, 1 form the alphabet.
 - ▼ E.g., $100110 =$
- In general, radix r representations use the first r chars in $\{0\dots9, A\dots Z\}$ and have the form $d_{n-1}d_{n-2}\dots d_1d_0$.
Summing $d_{n-1}\times r^{n-1} + d_{n-2}\times r^{n-2} + \dots + d_0\times r^0$ will convert to base 10. Why to base 10?