The Big Picture

CPSC 1050 – Chapter 1

Objectives

- Describe the course objectives
- Introduce the layers of a computing system
- Describe the concept of abstraction and its relationship to computing
- Look at the history of computer hardware and software

Course Objectives

- Giving you a solid, broad understanding of how a computing system works
- Developing an appreciation for and understanding of the evolution of modern computing systems
- Giving you enough information about computing so that you can decide whether you wish to pursue the subject further

Course Topics

- History of Computer Science
- Information and Data Representation
- Computer Hardware and Architecture
- Algorithmic Problem Solving
- An Introduction to Programming (C++)
- Operating Systems

- Networks
- Applications (Spreadsheets, Database)
- World Wide Web
- Computer Security

We will study 16 chapters!

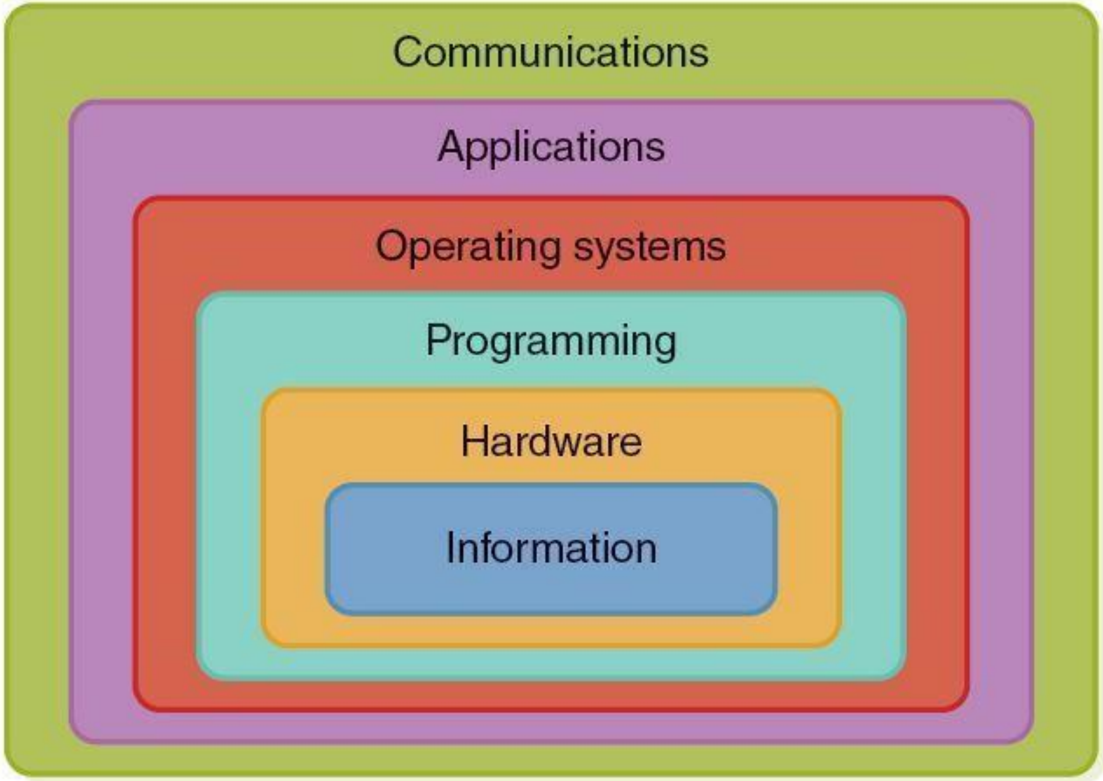


FIGURE 1.1 The layers of a computing system

Chapter 1
- History

Chapters 2-3
- Data as 1s, 0s

Chapters 4-5
- Chips, Circuits

Chapters 6-9
- Instructions

Chapters 10-11
- Linux, Windows

Chapter 12
- Excel, Database

Chapters 15-16
- WWW, Networks

Chapters 17-18
- Security, Limitations

We have a lot to learn!

- Let's make it simpler!
  - You want to drive. Do you need to know how an engine works?
    - You are a data entry clerk. Do you need to know programming? ○ You want to write a program. Do you need to know hardware?
- This is called Abstraction that focuses on external view and hides complex details
  - for example, when we run an application, we don't have to be concerned with how the program was written
- Some examples:
  - Texting on a smartphone ○ Surfing on the Internet ○ Printing a document

What is a Computing System?

- A Computing System is
  - is not just a single computer o is a dynamic entity that solves problems and interact with its environment
  - is often connected to other systems and Input/Output devices (also called I/O devices or Peripherals)
  - performs calculations, stores and processes data, and makes logical decisions

What is a Computing System?

- Computing System is composed of:
  - Hardware is the collection of physical elements that make up the machine and its related pieces (boxes, wires, circuit boards, keyboards, monitors, printers, …)

- Software is the collection of programs that provide the instructions for a computer to execute
- Data is the heart of a computing system

- Without data, the hardware and software are useless

What happened in the past?

History of Computers

Long time ago …

- Muhammad Al-Khwarizmi
  - Developed the concept of Algorithm
  - He was a Persian mathematician and astronomer of 9th century
- Do you think zero was always there?!
  - He introduced Algebra and the concepts of zero and negative numbers
  - NO!

- He is known as the "father of Algebra" and the "grandfather of computer science"

You will learn a lot of algorithms in this course!

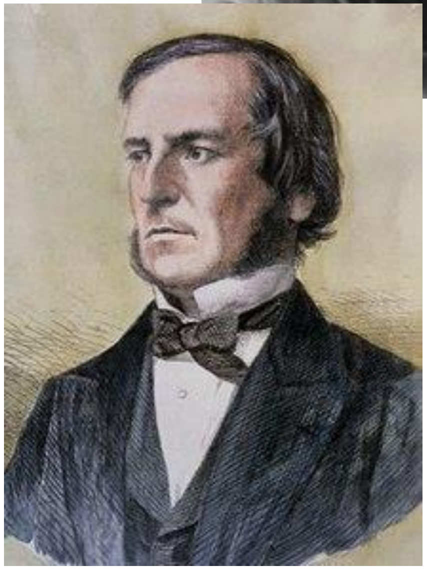
Later ...

- Binary System

  ○ base of computer systems: 0 and 1 ○ was developed by Gottfried Leibniz ○ The German philosopher in 17th century

- We also need Boolean Algebra

  ○ to represent logical circuits ○ George Boole developed this model ○ The British mathematician in 19th century



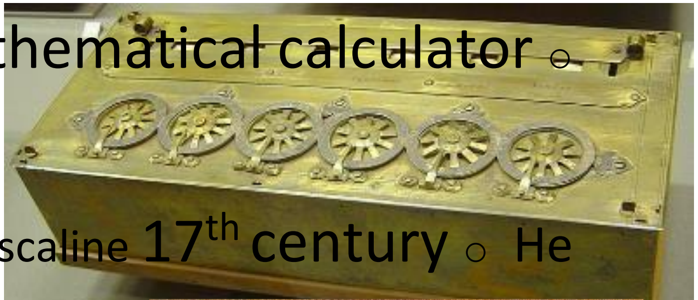
Let's start with Computational Devices!

- We have to jump back to 16th century!
- Have you ever seen this device?!

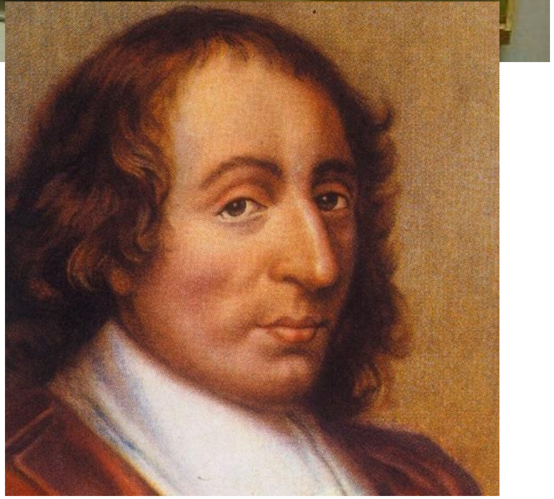
- It is called Abacus ○ An early device to record numeric values and perform basic arithmetic
- Do you know how it works?

Another Computational Device

Another Computational Device



- This device is called Pascaline or mathematical calculator ○ It could perform addition and subtraction
- It was invented by Blaise Pascal in Pascaline 17th century ○ He was a French mathematician



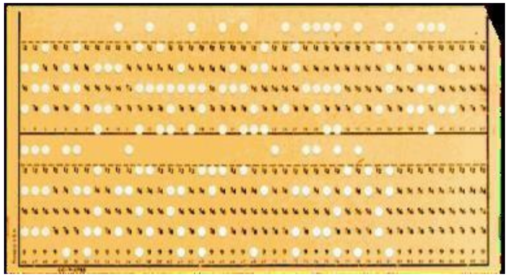
- Do you remember Leibniz?
  - ○ He later added multiplication feature to this device

What About Programming?

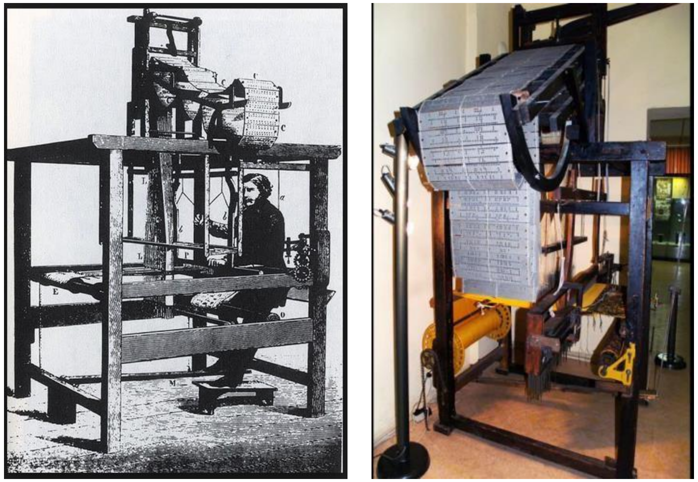
- Have you ever seen a punched card?

The punched card was used in early programming!

  - It is a perforated card



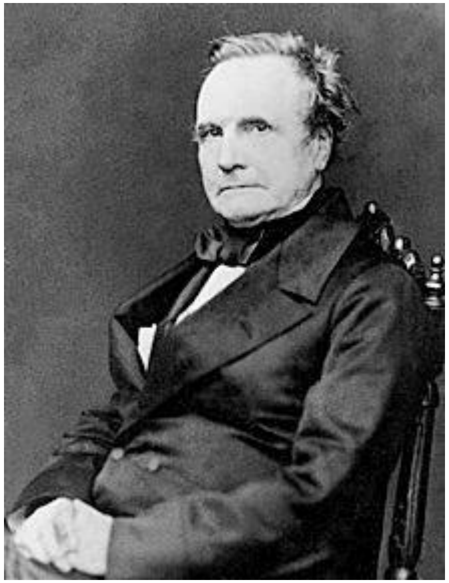
- Do you think it can represent 0 and 1?
- It was first used in 18th century by Joseph Jacquard in Jacquard's Loom for weaving cloth
  - It dictates the pattern into the cloth



Jacquard's Loom

First Steps into Computers

- Not much happened till 19th century
- Then, Charles Babbage, the British mathematician, designed Analytical Engine ○ first mechanical computer ○ included many of computer components such as memory and input ○ but too complex to be built!



He is called the "Father of Computer"

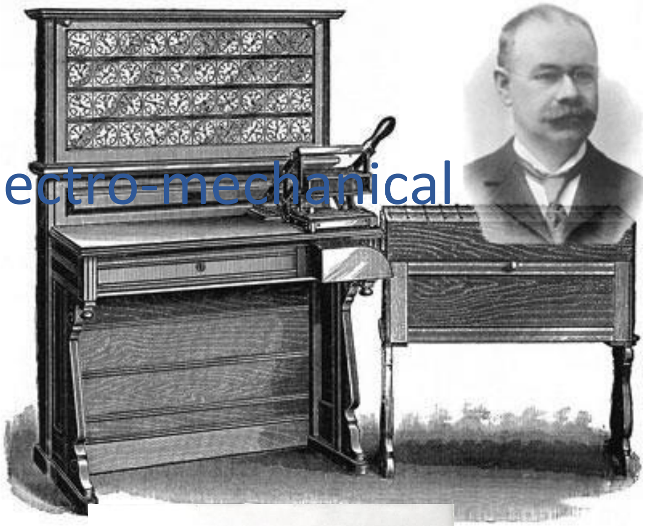
First Steps into Computers

Ada Lovelace was the first programmer!

- She was an English mathematician and writer
- She extended the ideas of Babbage
- She developed the concept of the loops
- There was a programming language named after Ada

Early Computers

- A lot more happened in the 20th century
- Herman Hollerith developed the first electro-mechanical tabulator
  - Designed to assist in summarizing information stored on punched cards
  - He formed the company IBM
- Alan Turing invented Turing Machine, a mathematical model
  - Foundation of computing theory
  - Code Breakers!



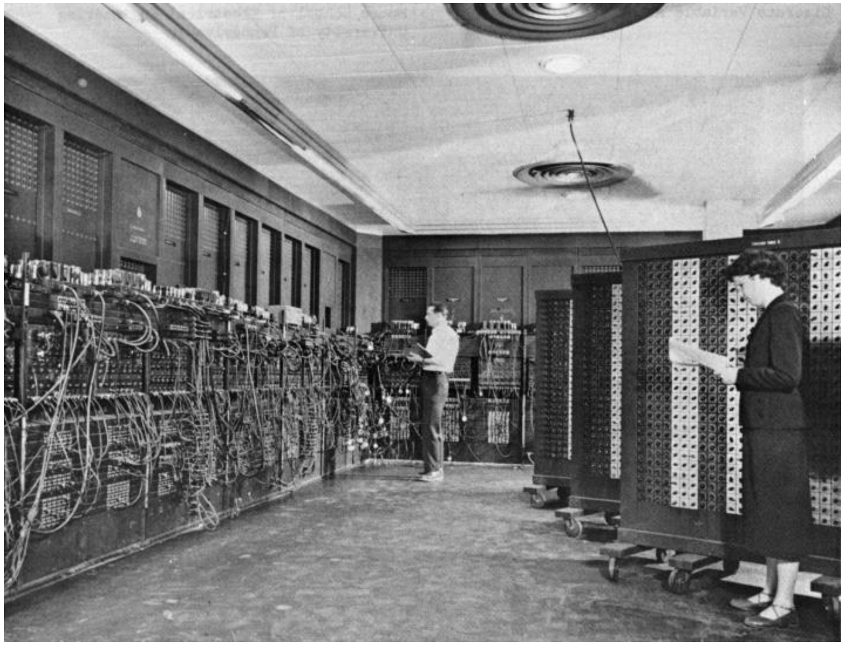
Can machines think?

Early Computers

  - First all-programmable computer
- Harvard Mark I (1944)
  - from IBM given to Harvard
- ENIAC (1946) / EDVAC (1950)
  - John von Neumann worked on both
- UNIVAC I (1951)
  - First commercial computer used for

Colossus (1943)

US Census



Eniac was the

- The era of early computers **size of a room!** ended at this point!

Early Computers

Thomas Watson, president of IBM, in 1943 predicted that:

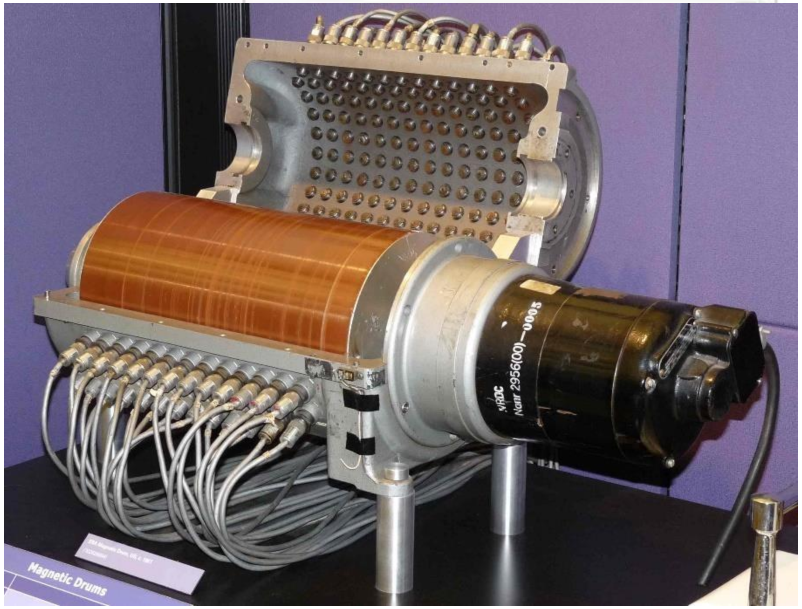
"I think there is a world market for maybe five computers!!"

- But a new era began in mathematics, physics, engineering, and economics
  - focus was not only on faster devices, but also on developing more productive components
- The following "generations" are based on the technologies they use

- first we talk about hardware generations ○ then we go on with software generations

HW 1st Generation (1951-1959)

- Vacuum Tube
  - used to generate electricity
  - large, not reliable, a lot of heat



- Magnetic Drum
  - rotated under a read/write head
  - used as memory

HW 1st Generation (1951-1959)

- Card Reader
  - was the input device

- Magnetic Tape Drive
  - sequential (?) storage device

HW 2nd Generation (1959-1965)

- Transistor

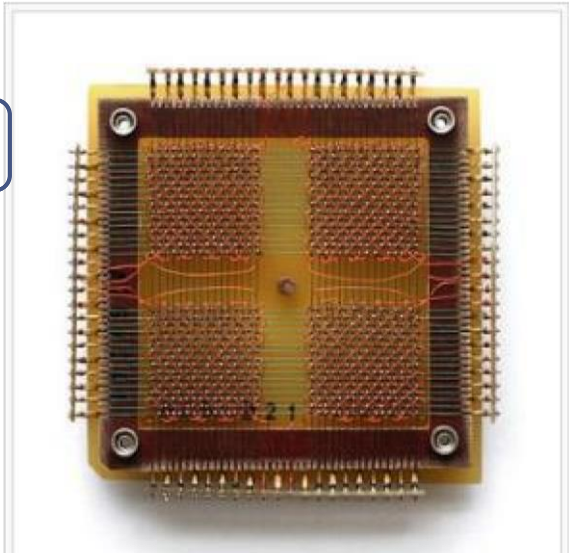
  replaced vacuum tubes

  - produces electricity
  - fast, small, durable, reliable, cheap

- Magnetic Core

  replaced magnetic drives

  - used as memory
  - info available instantly



HW 2nd Generation (1959-1965)

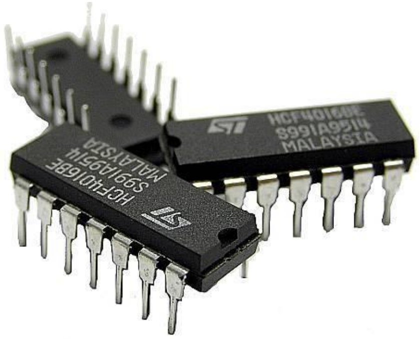
- Magnetic Disk
  - used as storage
  - data can be accessed directly (not sequential)

- Circuit Board
  - all components assembled on boards

HW 3rd Generation (1965-1971)

- Integrated Circuit (IC)

replaced circuit boards



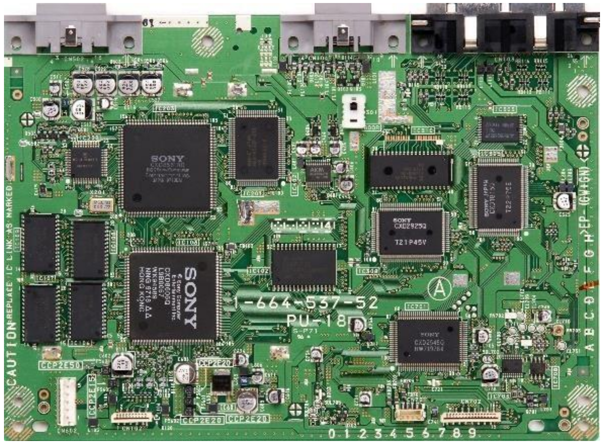
  - contained transistors and other components with their connections
  - smaller, cheaper, faster, more reliable
- Terminal
  - an input/output device with a keyboard and screen

Moore's Law

The number of circuits on a single IC was doubling each year

HW 4th Generation (1971-?)

- Large-Scale Integration (LSI) ○ build a whole microcomputer on one chip
- New companies emerge: Apple, Atari, Sun, Dell …
- Personal Computers (PC) and Workstations emerge
- Networking allowed computer interactions
- No end date as circuit boards are still used



Moore's Law

Computers will either double in power at the same price or halve in cost for the same power every 18 months

SW 1st Generation (1951-1959)

There are 3 types of Programming Languages:

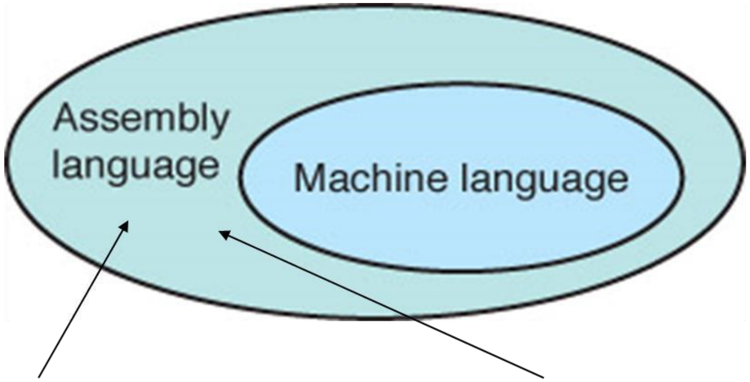
1. Machine Language

o Computer programs written in binary (1s and 0s)

2. Assembly Language

o English-like abbreviations (mnemonic codes) to write basic operations o

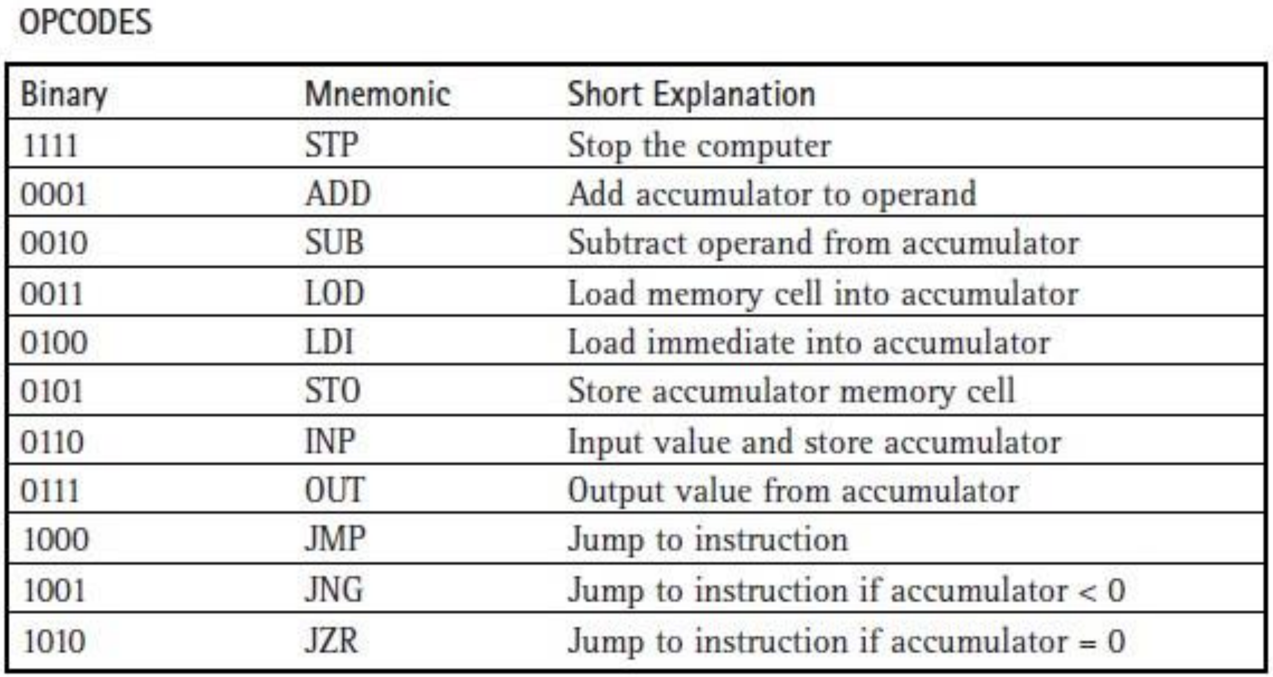
Translator programs (assemblers) convert assembly to machine language



Systems programmers *write* the assembler (translator)

Applications programmers *use* assembly language to solve problems

SW 1st Generation (1951-1959)



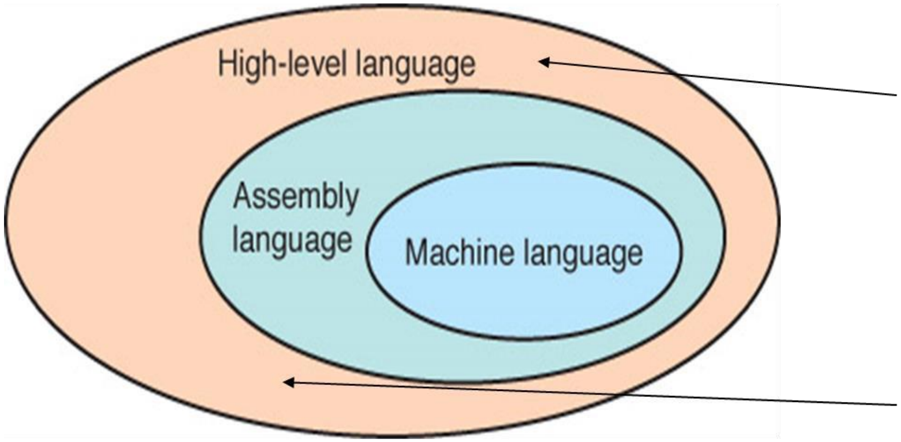
OPCODES

Binary	Mnemonic	Short Explanation
1111	STP	Stop the computer
0001	ADD	Add accumulator to operand
0010	SUB	Subtract operand from accumulator
0011	LOD	Load memory cell into accumulator
0100	LDI	Load immediate into accumulator
0101	STO	Store accumulator memory cell
0110	INP	Input value and store accumulator
0111	OUT	Output value from accumulator
1000	JMP	Jump to instruction
1001	JNG	Jump to instruction if accumulator < 0
1010	JZR	Jump to instruction if accumulator = 0

Super Simple CPU Instruction Set

SW 2nd Generation (1959-1965)

3. High-Level Language ○ English-like statements made programming easier ○ Fortran, COBOL, Lisp ○ Compiler converts high-level language into machine language



Systems programmers *write* translators for high-level languages

Application programmers *use* high-level languages to solve problems

SW 3rd Generation (1965-1971)

- Human operators slow down the computers

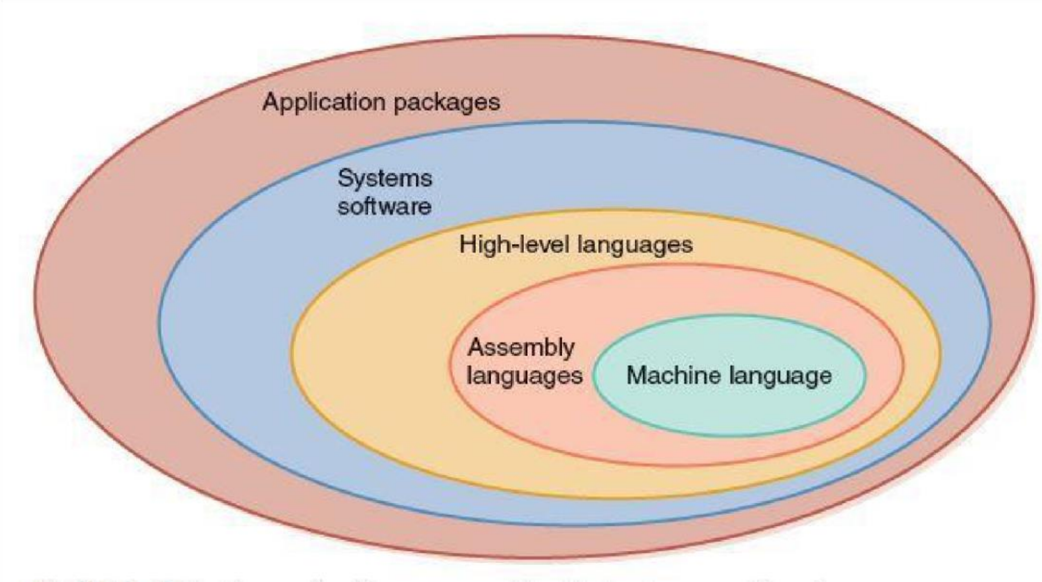
- Systems Software were created
  1. Utility programs perform common tasks
  2. Language translators (assemblers and compilers)
  3. Operating System controls the computers instead of humans

The other type of software is Application Software

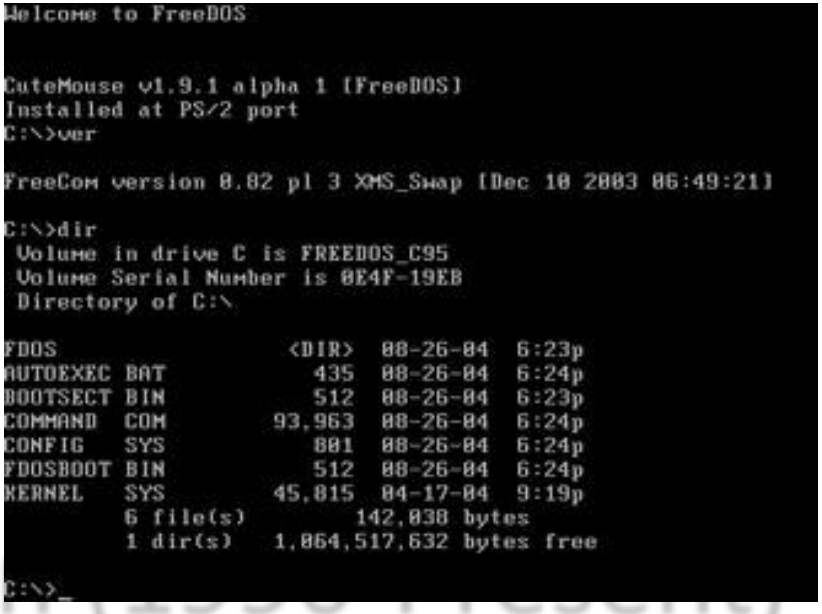
- Separation between user and programmer
  - Computer programmers write programs to be used by users

SW 4th Generation (1971-1989)

- Programming techniques improved: Structured Programming
  - A logical, disciplined approach to programming based on hierarchy of tasks
  - Pascal, C++



- More powerful operating systems developed
  - UNIX, DOS, graphical user interface by Apple
- High-quality Applications Software
  - Spreadsheets, Word processors, Database Management Systems



```
Welcome to FreeDOS

CuteMouse v1.9.1 alpha 1 [FreeDOS]
Installed at PS/2 port
C:\>ver

FreeCom version 0.82 pl 3 XMS_Swap [Dec 10 2003 06:49:21]

C:\>dir
 Volume in drive C is FREEDOS_C95
 Volume Serial Number is 0E4F-19EB
 Directory of C:\

FDOS                <DIR>  08-26-04   6:23p
AUTOEXEC BAT          435  08-26-04   6:24p
BOOTSECT BIN          512  08-26-04   6:23p
COMMAND  COM       93,963  08-26-04   6:24p
CONFIG   SYS          801  08-26-04   6:24p
FDOSBOOT BIN          512  08-26-04   6:24p
KERNEL   SYS       45,815  04-17-04   9:19p
         6 file(s)        142,038 bytes
         1 dir(s)   1,064,517,632 bytes free

C:\>_
```

SW 5th Generation (1990-Present)

- Rise of Microsoft
  - Windows operating system and other Microsoft application programs dominate the market

- Object-Oriented Design

○ used for large programming projects

The other type is

structured programming

○

based on a hierarchy of objects (e.g. Java)

- World Wide Web

○ allows easy global communication through the Internet

- New users

○ today's user needs no computer knowledge

Summary

- We start a broad study of computer systems including ○

Hardware that makes up the devices ○ Software programs executed by the machine ○ Data managed and manipulated by both

- The history of computing reveals the roots from which modern computing systems grew
- The rest of the book examines the different layers of a computing system

Additional Resources

- To Read!
  - A Brief History of Computer Science
  - The very Brief History of Computer Science
- To Watch!
  - Early Computing
  - Electronic Computing

The videos on this Crash Course Computer Science are recommended!