Introduction
- revolutions: agricultural, industrial, information
- 5-10% of the gross national products of USA
- innovation and progress
- double microprocessor performance every 18 months and double microprocessor complexity (transistors per chip) every 24 months
- interdependencies among assembly languages, organization, and design
- the boundary is where compilation (in software) ends and interpretation (in hardware) begins
- the instruction set architecture (architecture): interface between the hardware and the lowest-level software
- abstraction or hierarchical layers: lower-level details are hidden to offer a simpler model at higher levels
- implementation: the hardware that obeys the architecture abstraction

Below Your Program
- electrical signals: on and off
- binary numbers: 0 and 1
- bit: binary digit
- using numbers for both instructions and data
- low-level languages: machine-like
  - machine dependent
  - machine language
  - assembly language (mnemonic or symbolic)
  - assembler: translate symbolic notation to binary
- high-level programming languages: human-like, algebraic notation
  - machine independent
  - conciseness or expression power
  - compiler: translate high-level language to low-level
  - interpreter
  - very high-level languages: what instead of how
- subroutine libraries
- operating systems: resources management
- hierarchical layers of software and hardware
  - hardware
  - systems software: OS, compiler, assemblers
  - applications software

Components of Computer
- input device:
  - mouse: a pointing device, a large ball that contacts with a pair of wheels, one positioned on the x-axis and the other on the y-axis
- output device
  - graphic display:
    - picture element (pixel), matrix of bits (bit map) range from 512x340 to 1560x1280, b/w, gray scale, 3 primary colors (red, blue, green)
    - a raster refresh buffer, or frame buffer, to store the bit map; the image to be represented on-screen is stored in the frame buffer, and the bit pattern per pixel is read out to the graphics display at the refresh rate
    - raster cathode ray tube (CRT) display with fresh rate 30-75 times per second
    - liquid crystal display (LCDs), thin and low-power, the active matrix LCD has a tiny switch at each pixel to precisely control current and thus make sharper images
- main/primary memory: is where the programs and data are kept when they are running, volatile memory
- dynamic random access memory (DRAM), access time 50-100 nanoseconds
- cache
- central processor unit (CPU):
  - datapath: perform the arithmetic operations
  - control: command datapath, memory, I/O
- secondary memory: store programs/data between runs, nonvolatile
  - magnetic disk: a rotating platter coated with a magnetic recording material, access time 5-20 milliseconds, costs 50 times less than DRAM
    - hard disk, non-removable
      - a collection of platters, 5400-15000 revolutions per minute
      - movable arm with read/write head
    - floppy disk, flexible, removable, 1.44MB to 100MB Zip
    - optical compact disk (CD), cheaper, slower
    - magnetic tape: sequential access memory, for backup, access time in seconds

Computer Networks
- communication, resource sharing, non-local or remote access
- local area network: Ethernet
- wide area network and internet (www): optical fibers

Structured Computer Organization
structuring computers as a series of abstraction, each abstraction building on the one below it; the set of data types, operations, and features of each level is its architecture.

- **Languages and Virtual Machines:**
  - A machine defines a language: consists of all the instructions that the machine can execute.
  - A language defines a machine: the machine can execute all programs written in that language.
  - A whole series of languages (layers or levels), one on top of another, each one more convenient that its predecessors.
  - A computer with \( n \) levels can be regarded as \( n \) different virtual machines, each with a different machine language; a person write programs for a level \( n \) virtual machine need not be aware of the underlying interpreters and translators.

- **Contemporary Multilevel Machines:**
  - Level 0: digital logic (or gate) level, which is on top of the device (or transistor) level, which in turn on top of solid-state physics.
  - Level 1: microarchitecture level; the datapath can be directly controlled by hardware or controlled by microprogram (an interpreter for the instructions, it fetches, examines, and executes instruction one by one, using the datapath to do so).
  - Level 2: instruction set architecture (ISA) level; the machine's instruction is carried out interpretively by the microprogram or hardware execution circuit.
  - Level 3: operating system machine level; below and on this level is done by system programmers and always interpreted; above this level is done by average programmers and usually by translation.
  - Level 4: assembly language level.
  - Level 5: problem-oriented language level.

- **Evolution of Multilevel Machines:**
  - Hardware and Software:
    - Hardware: electronic circuit, memory, and input/output device, etc.
    - Software: algorithms, programs, etc.
  - Hardware and software are logically equivalent: any operation performed by software can also be built directly into the hardware; any instruction executed by the hardware an also be simulated in software; the decision to put certain functions in hardware and others in software is based on cost, speed, reliability, flexibility, etc.
  - Microprogramming:
    - Early computers have 2 levels: ISA level and digital logic level.
    - Add microprogramming level to become 3 level computers: microprograms have a limited instruction set, instead of ISA-level programs, which have a much larger instruction set, fewer electronics circuits would be needed.
    - Migration of functionality to microcode.

- **Integrated Circuit:**
  - Conductor, insulator, and semiconductor technology.
  - The 0\(^{th} \) generation: mechanical computers: Blaise Pascal 1642 (addition and subtraction), Baron G.W. von Leibniz (multiply and division), Charles Babbage (speedometer, difference engine, writing copper engraver's plate, analytical engine with store, mil, input, output, and first programmer Ada A. Lovelace), Konrad Zuse (electromagnetic relays), John Atanasoff and George Stibitz (binary numbers, capacitors memory), Howard Aiken (relays, Mark I & II).
  - The 1\(^{st} \) generation: vacuum tube.
  - Transistor: an on/off switch controlled by electricity.
  - Integrated circuit.
  - VLSI or very large-scale integrated circuit.
  - DRAM capacity quadrupled every 3 years.
  - Chip manufacturing process: silicon ingot, wafers, patterns, dies/chips, bonding/packaging.
  - Defect and yield rate, cost/performance trade-offs:
    - Cost per die = cost per wafer / (dies per wafer * yield).
    - Dies per wafer = wafer area / die area.
    - Yield = 1 / (1 + (defects per area * die area / 2))
  - DRAM chips have significantly increased in die size with each generation, yet yield in 43% - 48%.

- **Fallacies and Pitfalls:**
  - Fallacies – some common held misconceptions.
  - Pitfalls – easily made mistakes, and often are generalization of principles that are in a limited context.
  - Fallacy: computers have been built in the same old-fashioned way for far too long, and this antiquated model of compution is running out of steam.
    - CPU performance doubling every 18 months.
  - Pitfall: ignoring the inexorable progress of hardware when planning a new machine.
    - Many project are canceled because the project is late such that the performance is below the industry average.
Historical Perspective

- special-purpose programmable computer – late 1930 or late 1940
- Colossus – special-purpose decrypt computer, 1943
- ENIAC – the first operational electronic, general-purpose computer, 1946
- magnetic core memory - 1947
- Mark-I – the first stored-program or von Neumann computer, 1948
- Mark-III – separate memories for instructions and data
- EDSAC – the first full scale, stored-program computer, 1949
- UNIVAC I – the first commercial general-purpose computer, 1951
- IBM 701 – the first IBM commercial computer, 1952
- CDC 6600 – the first supercomputer, 1963
- IBM System/360 – 1964
- DEC PDP-8 – the first commercial minicomputer, 1965
- Intel 4004 – the first microprocessor or embedded processor, 1971
- Cray-1 – 1976
- Apple II – the first personal computer, 1977
- IBM personal computer – 1981