Computer Systems Architecture

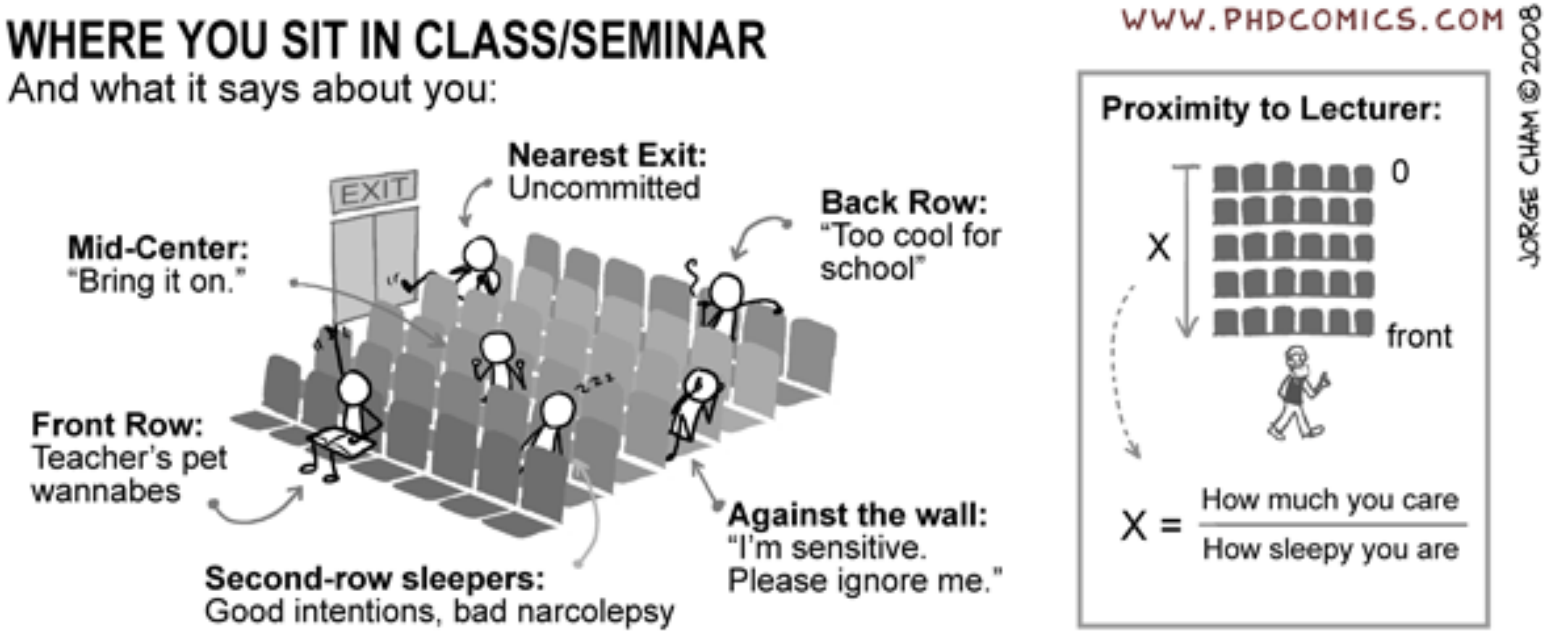
Fall 2012

Huzefa Rangwala, PhD

http://www.cs.gmu.edu/~hrangwal

*Slides adapted from Computer Organization and Design by Patterson and Henessey

* Credits to Dr. Arun Sood and Dr. Daniel Barbara for sharing their materials.


WHERE YOU SIT IN CLASS/SEMINAR
And what it says about you:
EXIT
Nearest Exit:
Uncommitted
Back Row:
"Too cool for school"
Mid-Center:
"Bring it on."
Front Row:
Teacher's pet wannabes
Against the wall:
"I'm sensitive. Please ignore me."
Second-row sleepers:
Good intentions, bad narcolepsy
WWW.PHDCOMICS.COM
JORGE CHAM © 2008
Proximity to Lecturer:
0
X
front
X = How much you care / How sleepy you are

Today ..

- Welcome
- Syllabus
  - Grading
  - Policies
  - Things you want to know and care about.
- Why should you be in this class ?
- What is computer architecture ?
- Demos

SYLLABUS

Location and Office Hours

- Class Time: TThu – 12:00 pm to 1:15 pm
- Class Location: Arts Building, L008
- Instructor: Huzefa Rangwala
  - Office Hours: Tuesday (2:30-4:30 pm) in Engineering Building #4423.
  - Email: rangwala@cs.gmu.edu
- TA: Azad Naik
- Office Hours: M-W(1:00-2:00 pm) in Engineering Building #4457
  - Email: anaik3@gmu.edu

Grading

- HW 0 0%
- HW 1 10% (Use of MIPS)
- HW 2 10% (Use of MIPS)
- HW 3 15% (Use of MIPS)
- HW 4 15% (Cache/MIPS)
- Mid-Term 20% (Closed Book)
- Final Exam 25% (Closed Book)
- Class Participation 5%
- Quizzes (Extra) 5%

Class Website and Announcements

- Keep Track of
  - http://www.cs.gmu.edu/~hrangwal/cs465-fall2012

- Specific Readings will be posted after each lecture. Note, the book is detailed and I will make clear what you are responsible for exams/assignments and also what is extra or "if-interested" reading.

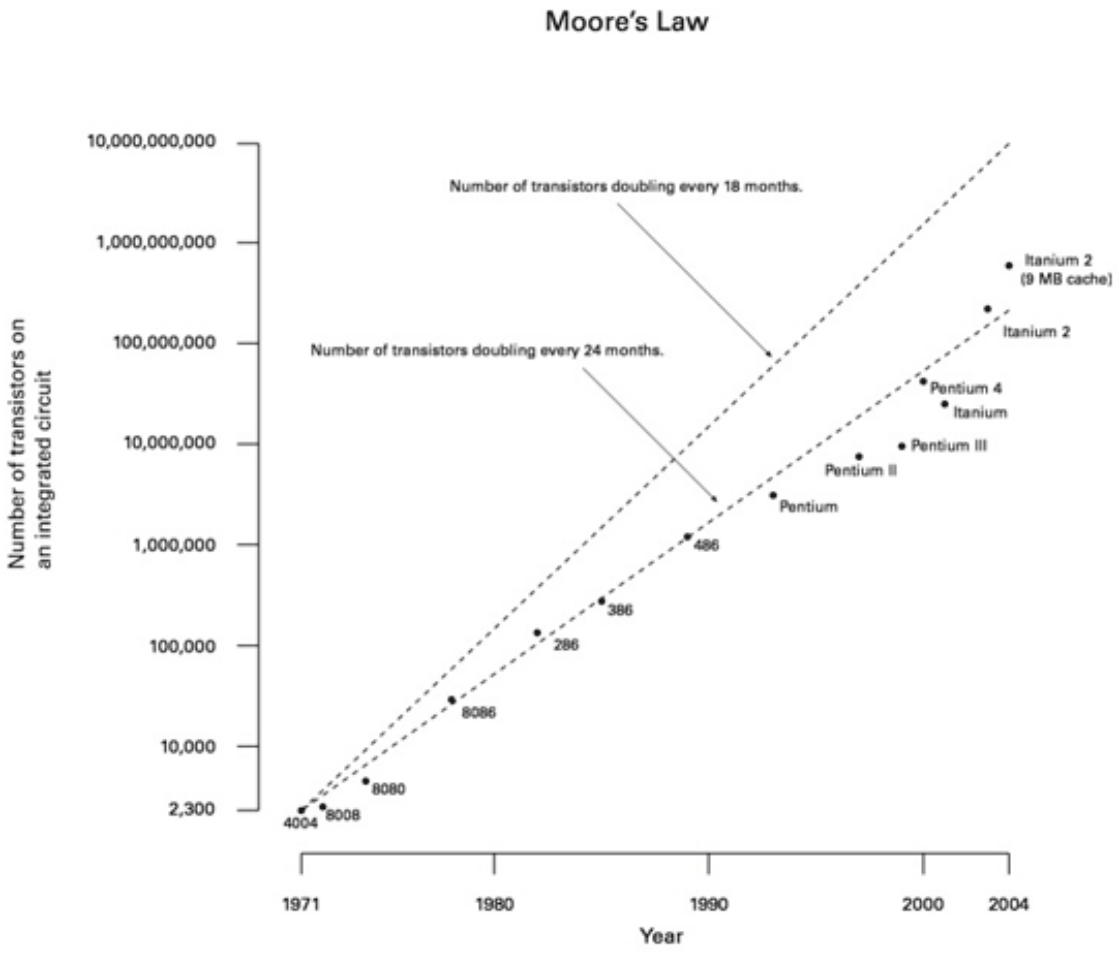
Write-Pair-Share Activity*

- What does computer architecture mean for you ?
  - Think about this individually and write 2-3 bullets. (2 minutes)
  - Meet your closest neighbor and discuss this with him/her (3 minutes)

Why? Besides it being required ….

- CS majors

Moore's Law


Moore's Law
Number of transistors doubling every 18 months.
Number of transistors doubling every 24 months.
Number of transistors on an integrated circuit
10,000,000,000
1,000,000,000
100,000,000
10,000,000
1,000,000
100,000
10,000
2,300
4004
8008
8080
8086
286
386
486
Pentium
Pentium II
Pentium III
Pentium 4
Itanium
Itanium 2
Itanium 2 (9 MB cache)
1971
1980
1990
2000
2004
Year

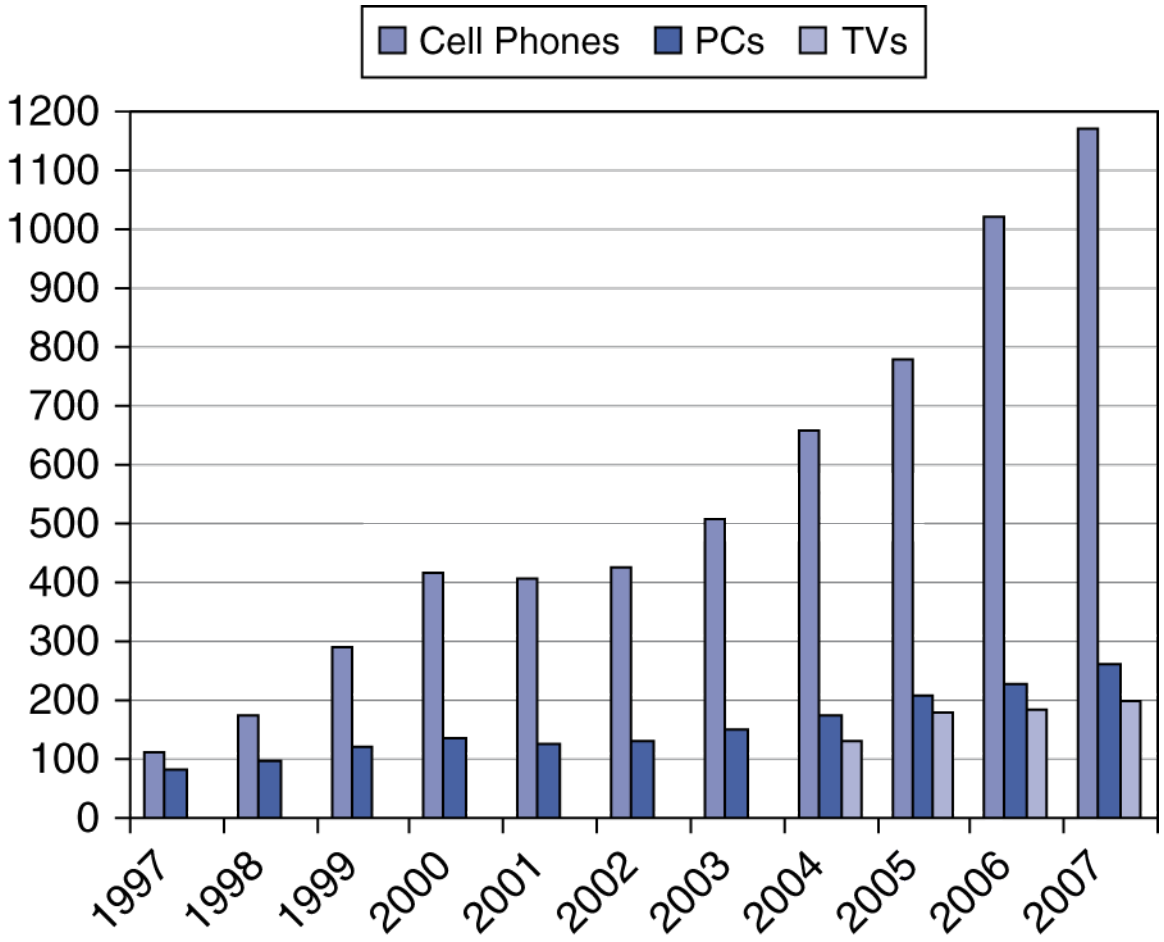
The Computer Revolution

- Progress in computer technology
  - Underpinned by Moore's Law
- Makes novel applications feasible
  - Computers in automobiles
  - Cell phones
  - Human genome project
  - World Wide Web
  - Search Engines
- Computers are pervasive

Classes of Computers

- Desktop computers
  - General purpose, variety of software
  - Subject to cost/performance tradeoff
- Server computers
  - Network based
  - High capacity, performance, reliability
  - Range from small servers to building sized
- Embedded computers
  - Hidden as components of systems
  - Stringent power/performance/cost constraints

The Processor Market



What You Will Learn

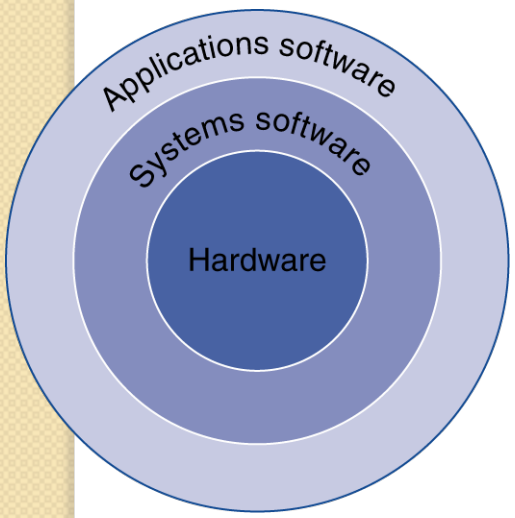
- How programs are translated into the machine language
  - And how the hardware executes them
- The hardware/software interface
- What determines program performance
  - And how it can be improved
- How hardware designers improve performance
- What is parallel processing

Understanding Performance

- Algorithm
  - Determines number of operations executed
- Programming language, compiler, architecture
  - Determine number of machine instructions executed per operation
- Processor and memory system
  - Determine how fast instructions are executed
- I/O system (including OS)
  - Determines how fast I/O operations are executed

Below Your Program

- Application software
  - Written in high-level language
- System software
  - Compiler: translates HLL code to machine code
  - Operating System: service code
    - Handling input/output
    - Managing memory and storage
    - Scheduling tasks & sharing resources
- Hardware
  - Processor, memory, I/O controllers



High-level to Machine Language

```
swap(int v[], int k)
{int temp;
   temp = v[k];
   v[k] = v[k+1];
   v[k+1] = temp;
}
```

High-level language program
(in C)

Compiler

```
swap:
    muli $2, $5,4
    add  $2, $4,$2
    lw   $15, 0($2)
    lw   $16, 4($2)
    sw   $16, 0($2)
    sw   $15, 4($2)
    jr   $31
```

Assembly language program
(for MIPS)

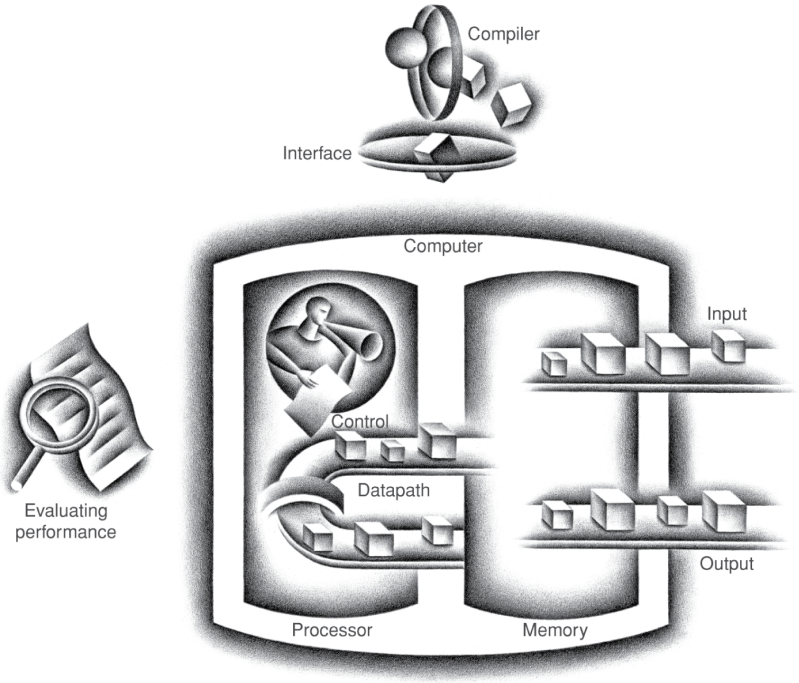
Assembler

```
00000001010001000000000000011000
00000001000111000011000001000001
10001100011000100000000000000000
10001100111100100000000000000100
10101100111100100000000000000000
10101100011000100000000000000100
00000011111000000000000000001000
```

Binary machine language program
(for MIPS)

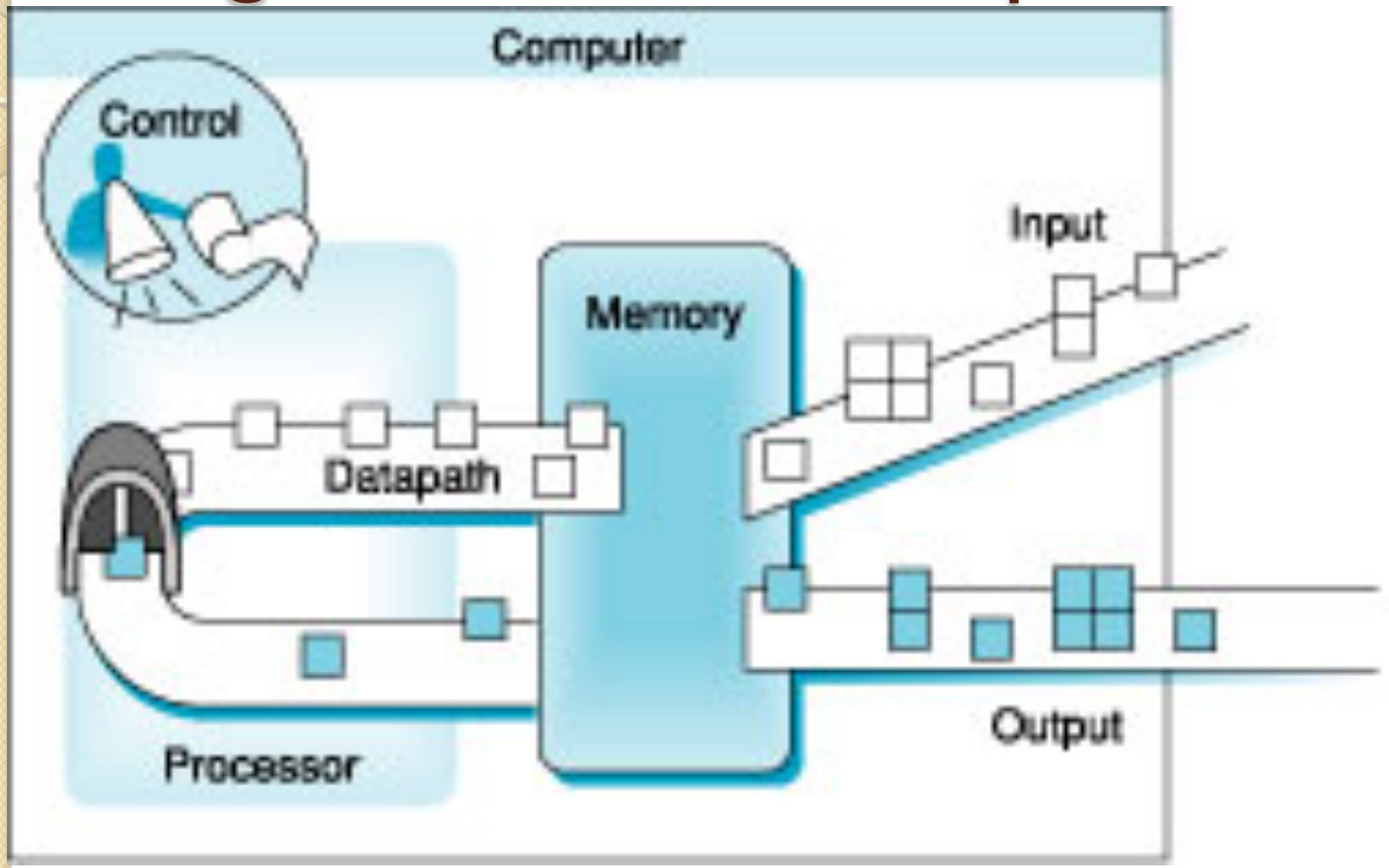
Components of a Computer

The BIG Picture



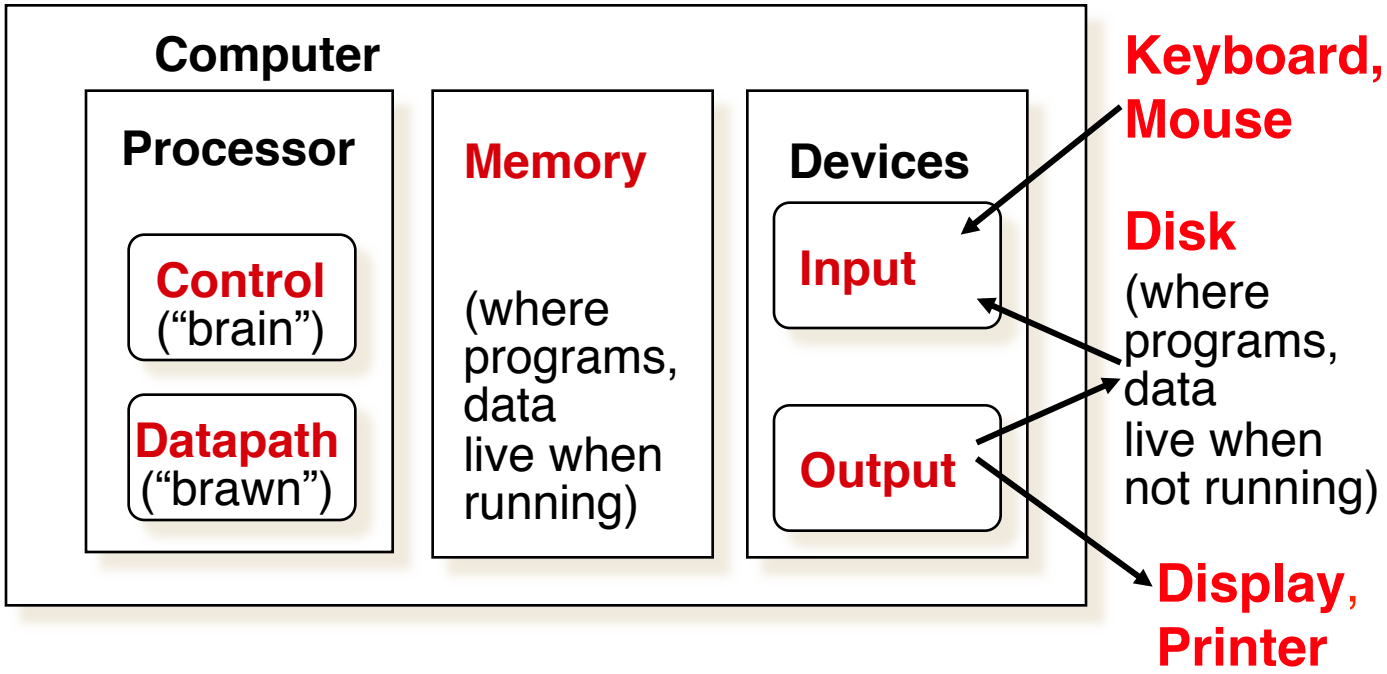
- Same components for all kinds of computer
  - Desktop, server, embedded
- Input/output includes
  - User-interface devices
    - Display, keyboard, mouse
  - Storage devices
    - Hard disk, CD/DVD, flash
  - Network adapters
    - For communicating with other computers

Organization of a computer



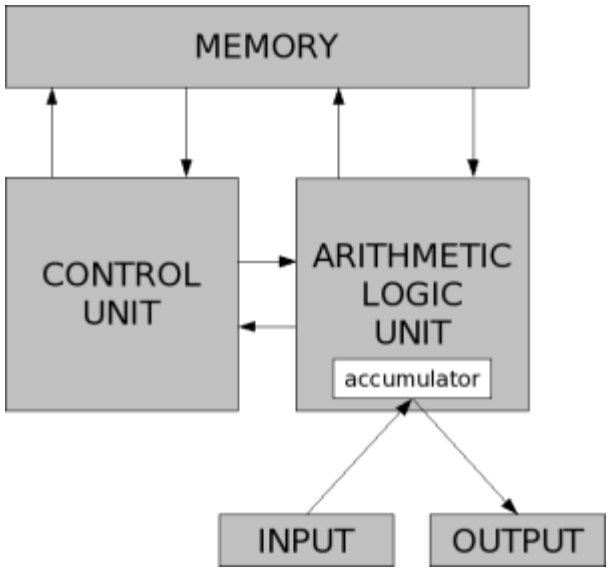
Anatomy of Computer

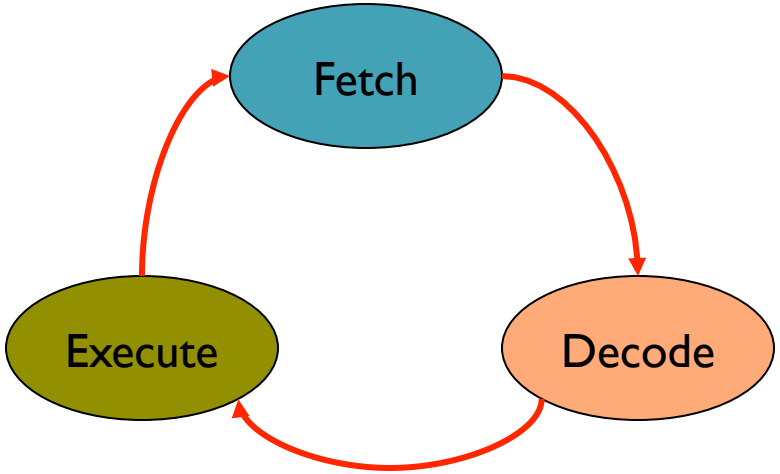
5 classic components



- Datapath: performs arithmetic operation
- Control: guides the operation of other components based on the user instructions

Von Neumman Architecture





From Wikipedia

The von Neumann architecture is a design model for a stored-program digital computer that uses a central processing unit (CPU) and a single separate storage structure ("memory") to hold both instructions and data

Why Learn Computer Architecture?

- You want to call yourself a "computer scientist"
  - Computer architecture impacts every other aspect of computer science
- You need to make a purchasing decision or offer "expert" advice
- You want to build software people use – sell many, many copies-(need performance)
  - Both hardware and software affect performance
    - Algorithm determines number of source-level statements
    - Language/compiler/architecture determine machine instructions (Chapter 2 and 3)
    - Processor/memory determine how fast instructions are executed (Chapter 4, 5, and 6)
    - Assessing and understanding performance (Chapter 1)
    - New Trends in Architecture: Multicores, Multiprocessors, GPUS (Chapter 7)

Demos –

- Demos
- Etc…