Levels of Abstraction

Computer Organization
Level of Abstraction

- Provides users with concepts/tools to solve problem at that level
- Implementation hides complex details
  - different implementations using different underlying “machines” are possible
Level of Abstraction -- Example

◆ Abstract Data Types
  – stack, queue, priority queue

◆ Use for building higher level programs
  – solution search (e.g. maze, TSP)
  – depth-first, breadth first, branch and bound

◆ Implementation
  – built on base language: C++, Pascal, Ada, etc.

◆ Can be used to build high level abstraction
  – generic solution search
  – event driven simulation language
Levels of Abstraction

Virtual Machine 3

- Implementation of VM 3 on VM 2

Virtual Machine 2

- Implementation of VM 2 on VM 1

Virtual Machine 1

- Implementation of VM 1 on VM 0

Virtual Machine 0
Data Structures example

- Simulation of Bank
- Event Scheduling Simulation Language
- Data Structures
- Base Language, e.g. C++
Hardware/Software Interface

simple RISC

1. Instruction Set Architecture (ISA)
   - Hardware

2. Operating System (OS)
   - Partial Interpretation (device drivers, etc)
   - Assembler (translation/mapping to ISA)

3. Assembly Language
   - Assembler (translation/mapping to ISA)

4. High Level Language (HLL)
   - Compiler (translation)

Hardware

Digital Logic

Electronics -- transistors, etc
Hardware/Software Interface
more complex architectures

1A
Microarchitecture

1
ISA

Partial Interpretation (device drivers, etc)

2
OS

Assembler (translation/mapping to ISA)

3
Assembler

Firmware interpreter

Hardware

Digital Logic

Electronics -- transistors, etc

Hardware/Electronics
Java

Java (HLL)

Java Compiler

Java Byte Code (ISA)

Interpreter and JIT compiler

Java Virtual Machine (JVM)

OS

Machine ISA
This Course

High Level Language (HLL)  COSC 420

Assembly Language  COSC 425

Operating System (OS)  COSC 301

Partial Interpretation (device drivers, etc)

Instruction Set Architecture (ISA)

Hardware

Digital Logic

Hardware

Electronics -- transistors, etc