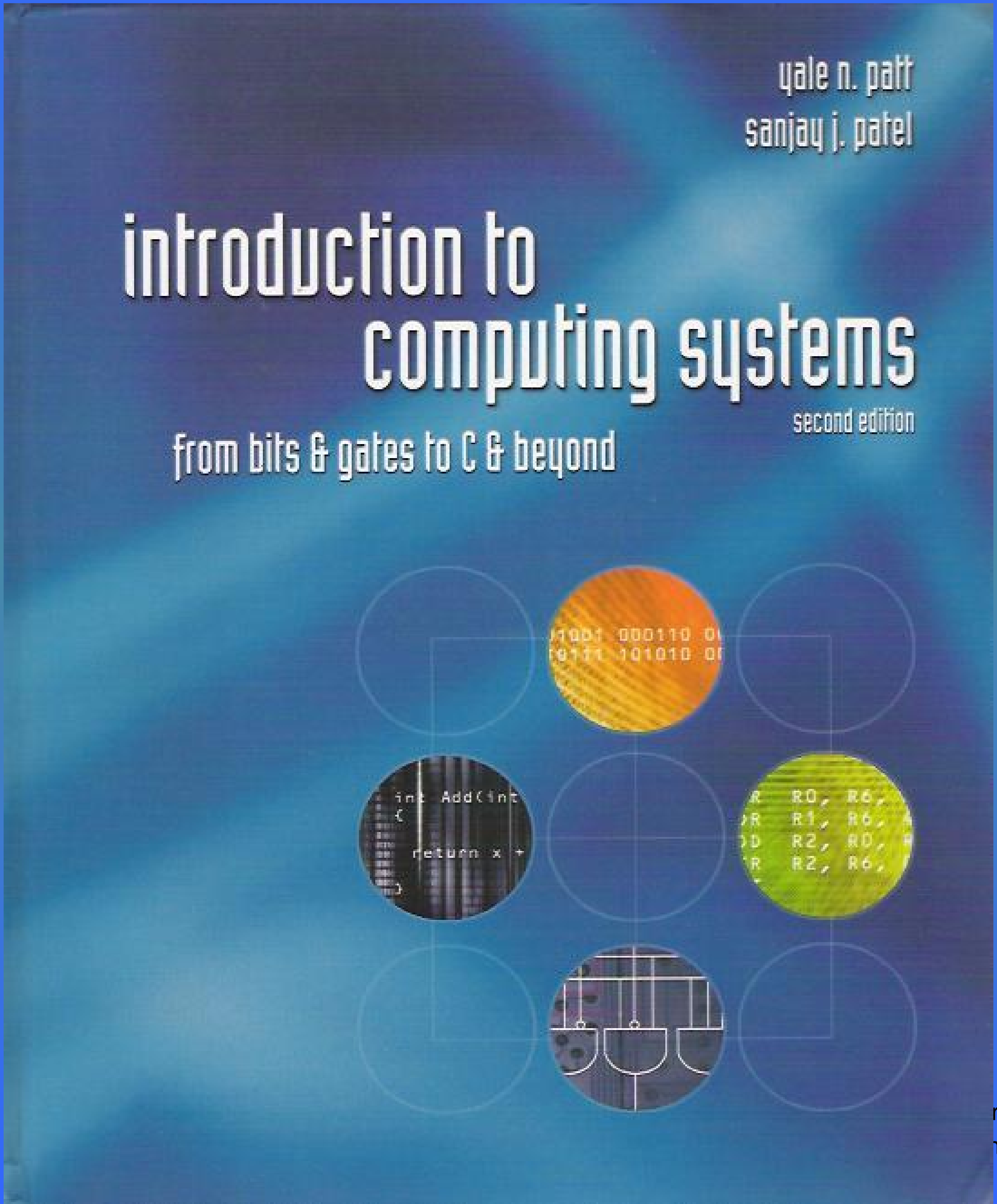
Mc
Graw
Hill



yale n. patt
sanjay j. patel

introduction to computing systems

second edition

from bits & gates to C & beyond

Introduction to Computing Systems
from bits & gates to C & beyond

Chapter 1

Welcome Aboard!

This course is about:

- What computers consist of
- How processors work
- How they are organized internally
- What are the design tradeoffs
- How design affects programming and applications

- ~~How to fix computers~~
- ~~How to build myself one real cheap~~
- ~~Which one to buy~~
- ~~Knowing all about the Intel core i5 or PowerPC~~

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

Computing Machines

- **Ubiquitous** (= everywhere)
  - General purpose: servers, desktops, laptops, PDAs, etc.
  - Special purpose: cash registers, ATMs, games, telephone switches, etc.
  - Embedded: cars, hotel doors, printers, VCRs, industrial machinery, medical equipment, etc.
- **Distinguishing Characteristics**
  - Speed (increasing)
  - Cost (decreasing)
  - Ease of use, software support & interface
  - Scalability

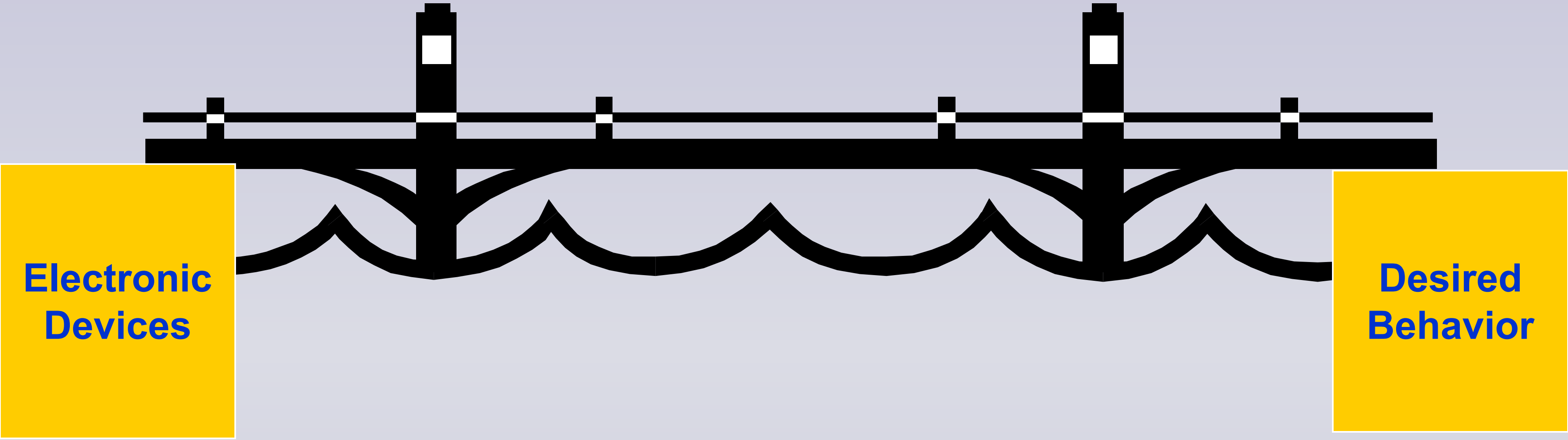
Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

Two recurring themes

- **Abstraction**
  - The notion that we can concentrate on one “level” of the big picture at a time, with confidence that we can then connect effectively with the levels above and below.
  - Framing the levels of abstraction appropriately is one of the most important skills in *any* undertaking.
- **Hardware vs. Software**
  - On the other hand, abstraction does *not* mean being clueless about the neighboring levels.
  - In particular, hardware and software are inseparably connected, especially at the level we will be studying.

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

What is Computer Organization?



Electronic Devices

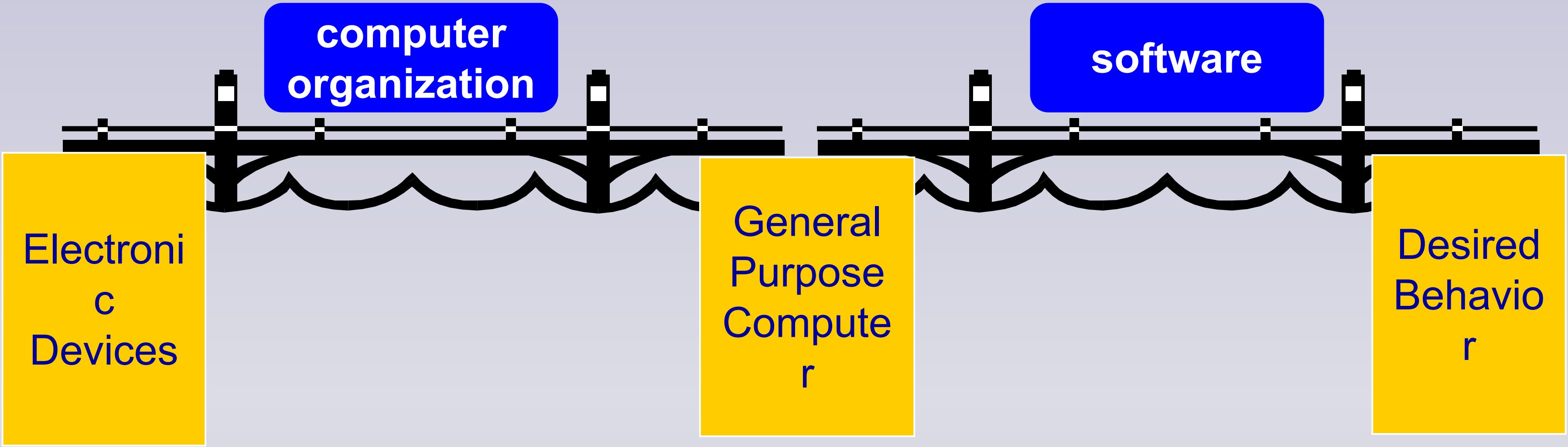
Desired Behavior

… a very wide “semantic gap” between the intended behavior and the workings of the underlying electronic devices that will actually do all the work.

The forerunners to modern computers attempted to assemble the raw devices (mechanical, electrical, or electronic) into a separate purpose-built machine for each desired behavior.

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

Role of General Purpose Computers



computer organization

software

Electronic Devices

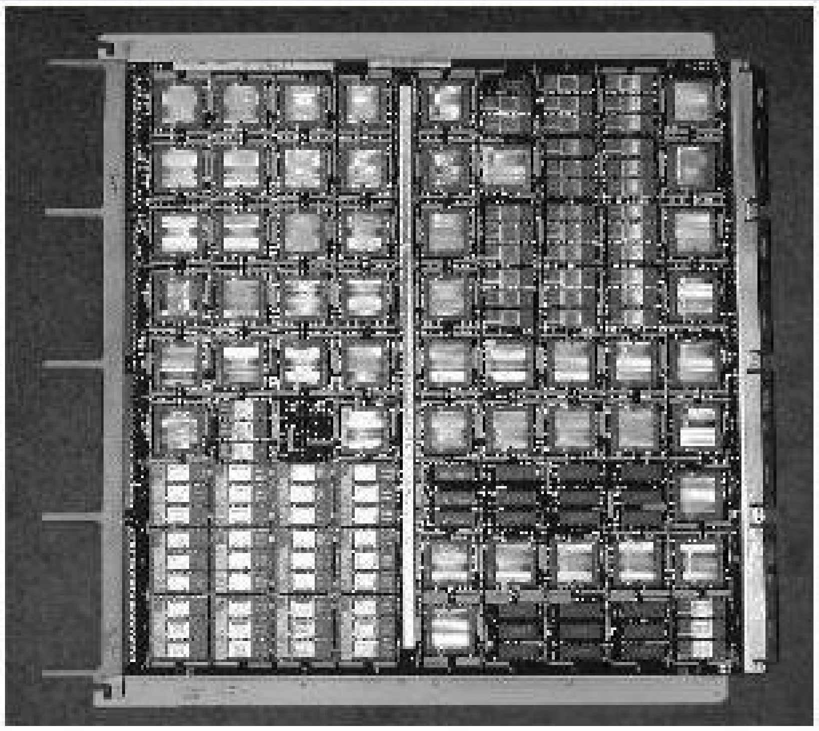
General Purpose Computer

Desired Behavior

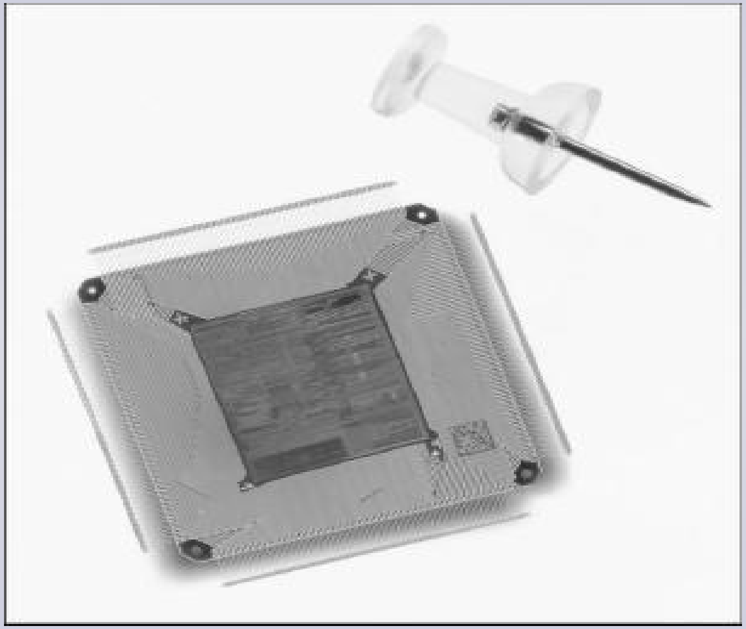
A general purpose computer is like an island that helps span the gap between the desired behavior (application) and the basic building blocks (electronic devices).

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

CPUs: the heart of computing systems



ca 1980
It took 10 of these boards to
make a *Central Processing Unit*



ca 2000
You can see why they called
this CPU a *microprocessor*!

Two pillars of Computing

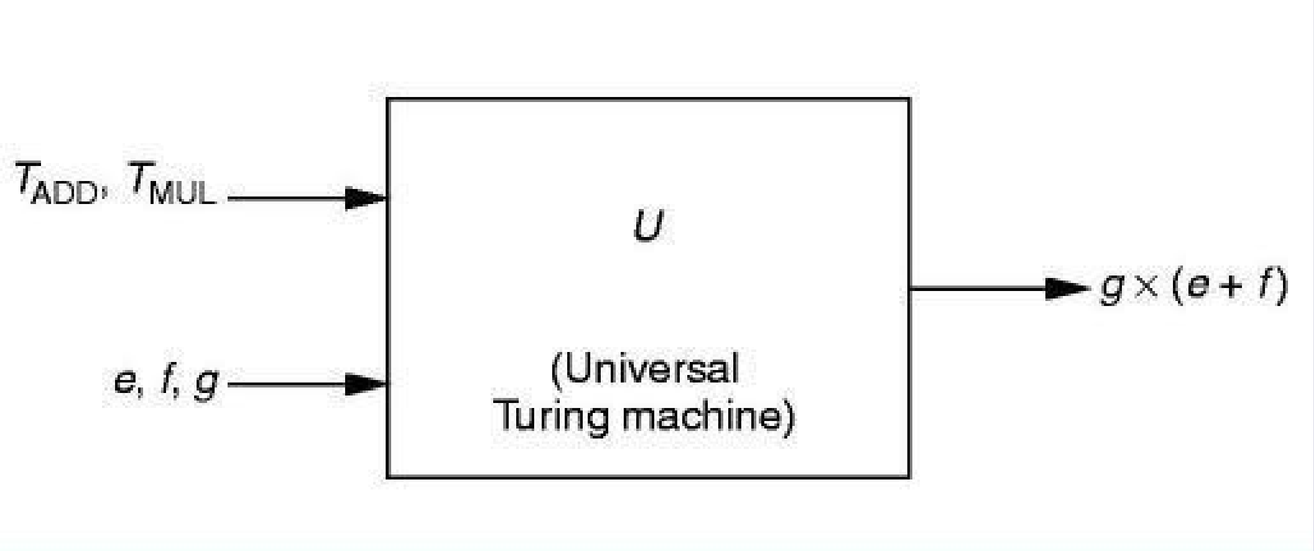
- **Universal Computational Devices**
  - Given enough time and memory, all computers are capable of computing exactly the same things (irrespective of speed, size or cost).
    - *Turing's Thesis: every computation can be performed by some "Turing Machine" - a theoretical universal computational device*

- **Problem Transformation**
  - The ultimate objective is to transform a problem expressed in natural language into electrons running around a circuit!
    - *That's what Computer Science and Computer Engineering are all about: a continuum that embraces software & hardware.*

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

A Turing Machine

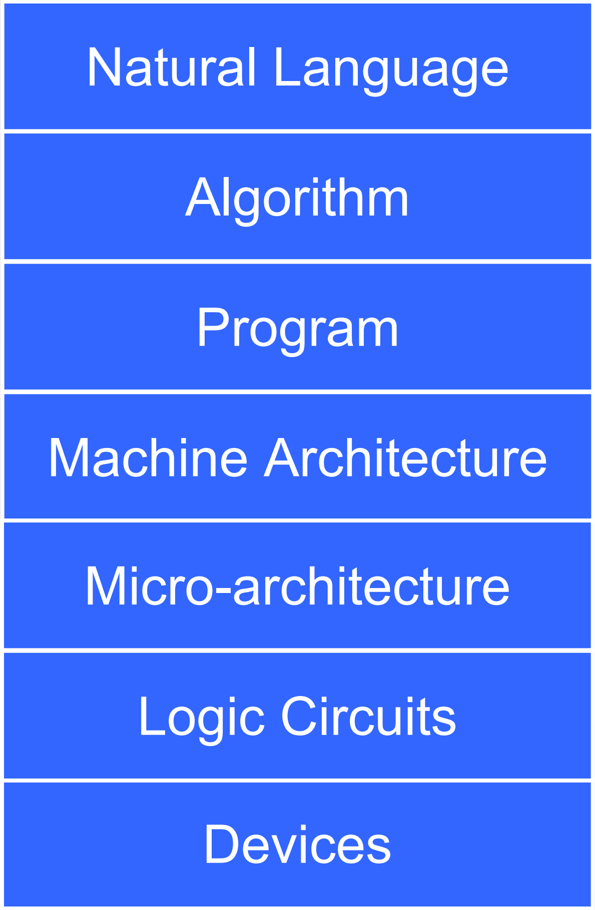
Also known as a *Universal Computational Device*: a theoretical device that accepts both input data and instructions as to how to operate on the data



Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

Problem Transformation - levels of abstraction

The desired behavior: the application



Natural Language

Algorithm

Program

Machine Architecture

Micro-architecture

Logic Circuits

Devices

The building blocks: electronic devices

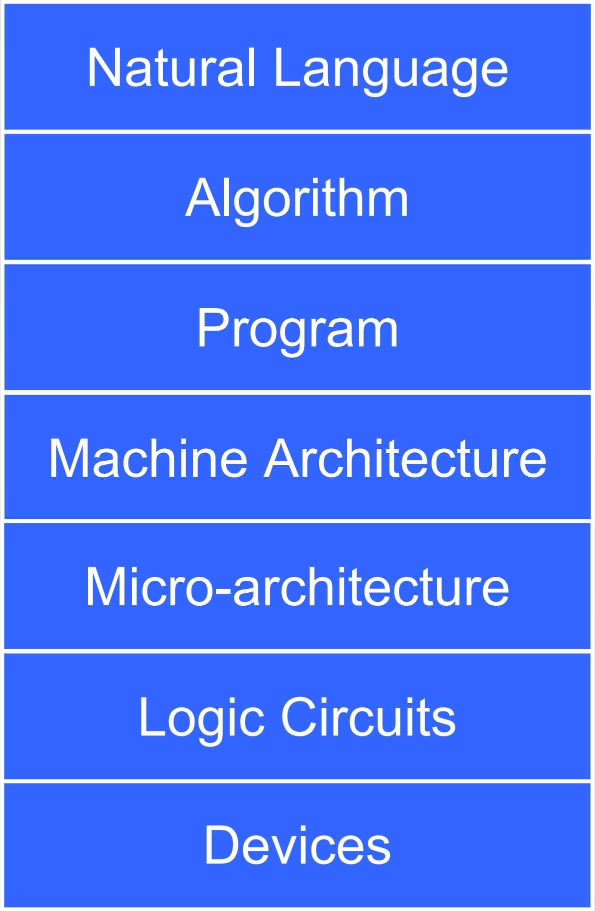
Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

Levels of Abstraction

- These levels do not necessarily correspond to discrete components, but to well defined *standard interfaces.*

- Standard interfaces provide
  - *portability*
  - *third party software/hardware*
  - *wider usage*

- These levels are to some extent arbitrary - there are other ways to draw the lines.



Natural Language

Algorithm

Program

Machine Architecture

Micro-architecture

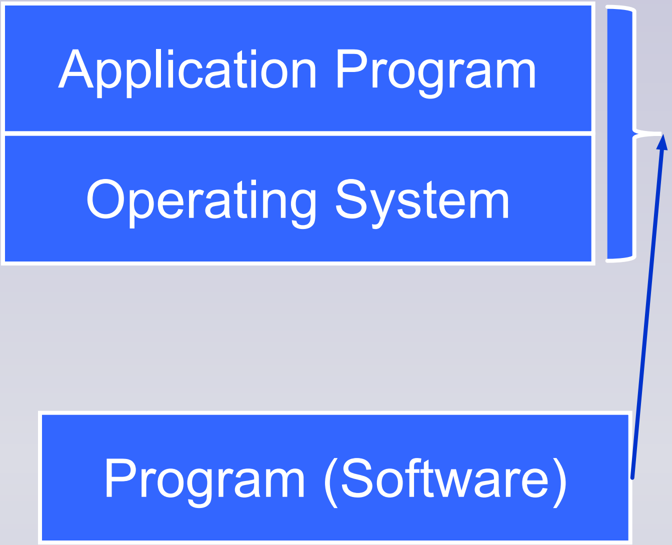
Logic Circuits

Devices

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

The Program Level

- **Most computers run a management program called the *operating system (OS).***
- **Application programs interface to the machine architecture via the OS.**



Application Program

Operating System

Program (Software)

An e

This lecture	Data
PowerPoint	Application Program
Windows XP	Operating System

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

The Machine Level - 1

- **Machine Architecture**
  - This is the formal specification of all the functions a particular machine can carry out, known as the *Instruction Set Architecture* (ISA).
- **Microarchitecture**
  - The implementation of the ISA in a specific CPU - i.e. the way in which the specifications of the ISA are actually carried out.

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

The Machine Level - 2

- **Logic Circuits**
  - Each functional component of the microarchitecture is built up of circuits that make “decisions” based on simple rules

- **Devices**
  - Finally, each logic circuit is actually built of electronic devices such as CMOS or NMOS or GaAs (etc.) transistors.

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside

Course Outline - What is Next?

- How to represent information for a computer
- The building blocks of computers: logic gates and logic circuits; memory circuits
- The basic algorithm: the von Neumann model
- An example: the LC-3 structure and language
- Programming the machine: assembly language
- A higher-level language: C

Slides prepared by Walid A. Najjar & Brian J. Linard, University of California, Riverside