

# 计算机系统

## 0. 课程简介

华东师范大学 数据科学与工程学院

2022年09月05日

钱卫宁

[wngqian@dase.ecnu.edu.cn](mailto:wngqian@dase.ecnu.edu.cn)

	第一学期	第二学期	一暑期	第三学期	第四学期	二暑期	第五学期	第六学期	第七学期	第六、七、八学期，三暑期
基础/ 导论	线性代数	高等数学(2)		<b>DaSE 导论</b>						选修课： 深度学习； IR+NLP； CV+MM； 高级DBMS； 事务处理； 智慧城市； 社会计算； 区块链； 设计思维； 电子商务...  实习实践 毕业论文
	高等数学(1)			概率论						
	程序设计									
计算机理论		数据结构		算法设计与分析			<b>DaSE 算法</b>			
		离散数学								
系统				计算机系统	操作系统		<b>当代 DBMS</b>	<b>分布计算 系统</b>		
								<b>云计算 系统</b>		
实践		Web编程	数据可视化		数据伦理		计算机网络原理与编程		软件工程	
模型与方法					<b>DaSE 数学</b>		<b>统计方法与ML</b>	<b>当代AI</b>		

# Welcome Quiz!

1. 请把十进制数 70 和 1.5 写成十六进制形式。
2. 请问集成开发环境（比如Visual C++或code blocks或VSCode）在编译你的程序时做了哪些事情？
3. 你日常使用的计算机操作系统是什么？如何知道系统中在运行哪些程序？
4. 你写过的最长的程序大约有多少行代码？包含几个源文件（含头文件）？

# 为什么需要深入理解计算机系统？

```
#include <stdio.h>

int main()
{
    printf("hello, world\n");
}
```

- 源代码是如何成为可运行程序的？
- 程序是如何运行的？

# 演示

```
$ gcc hello.c -o hello  
$ ./hello
```

```
$ gcc -E hello.c -o hello.i  
$ gcc -S hello.c -o hello.s  
$ gcc -c hello.s -o hello.o  
$ ld...
```

# 为什么需要深入理解计算机系统？

## 编译系统工作过程概览

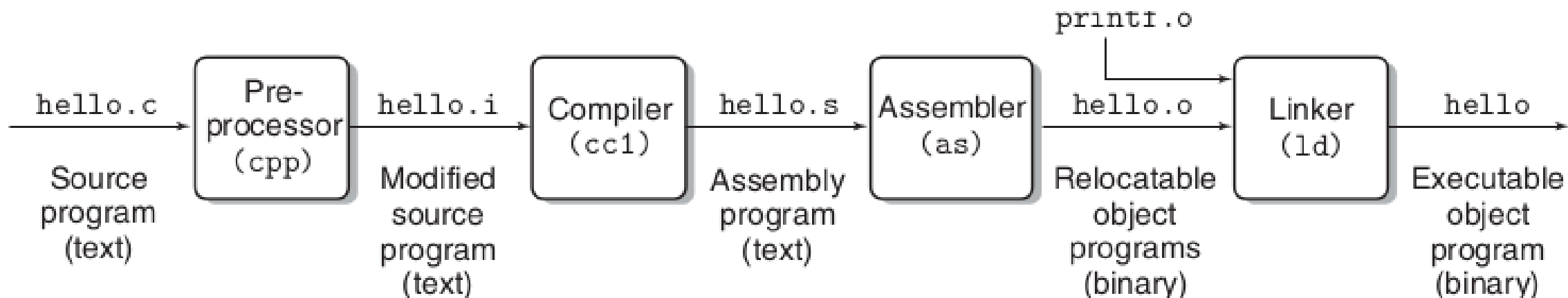


Figure 1.3 The compilation system.

# 为什么需要深入理解计算机系统？

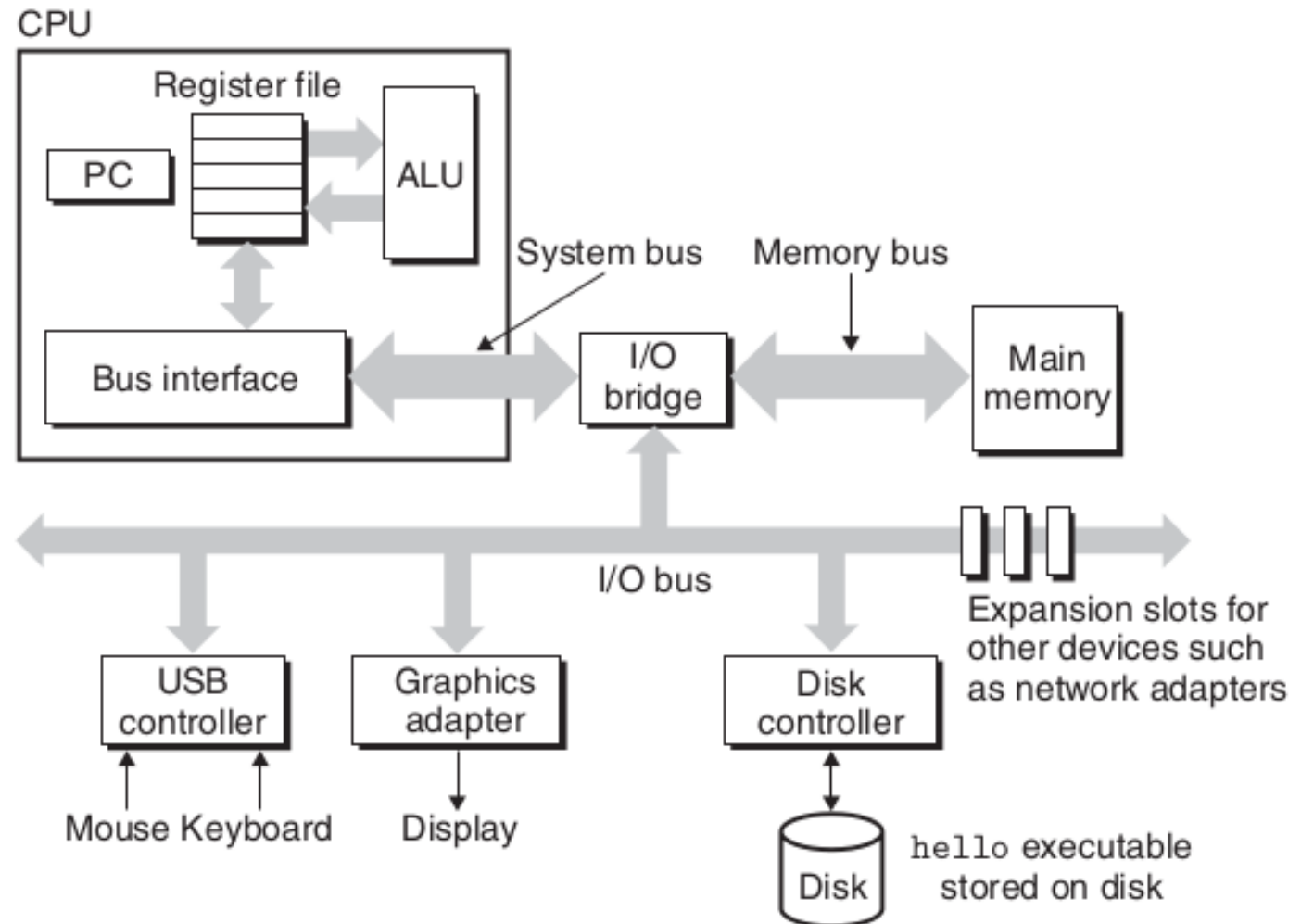
- 优化程序性能
- 理解各种错误
- 避免安全隐患

提高开发、设计的效率和质量

# 课程主要内容：一台抽象的“计算机”

Figure 1.4

**Hardware organization of a typical system.** CPU: Central Processing Unit, ALU: Arithmetic/Logic Unit, PC: Program counter, USB: Universal Serial Bus.





# 计算机如何表示代码和数据？

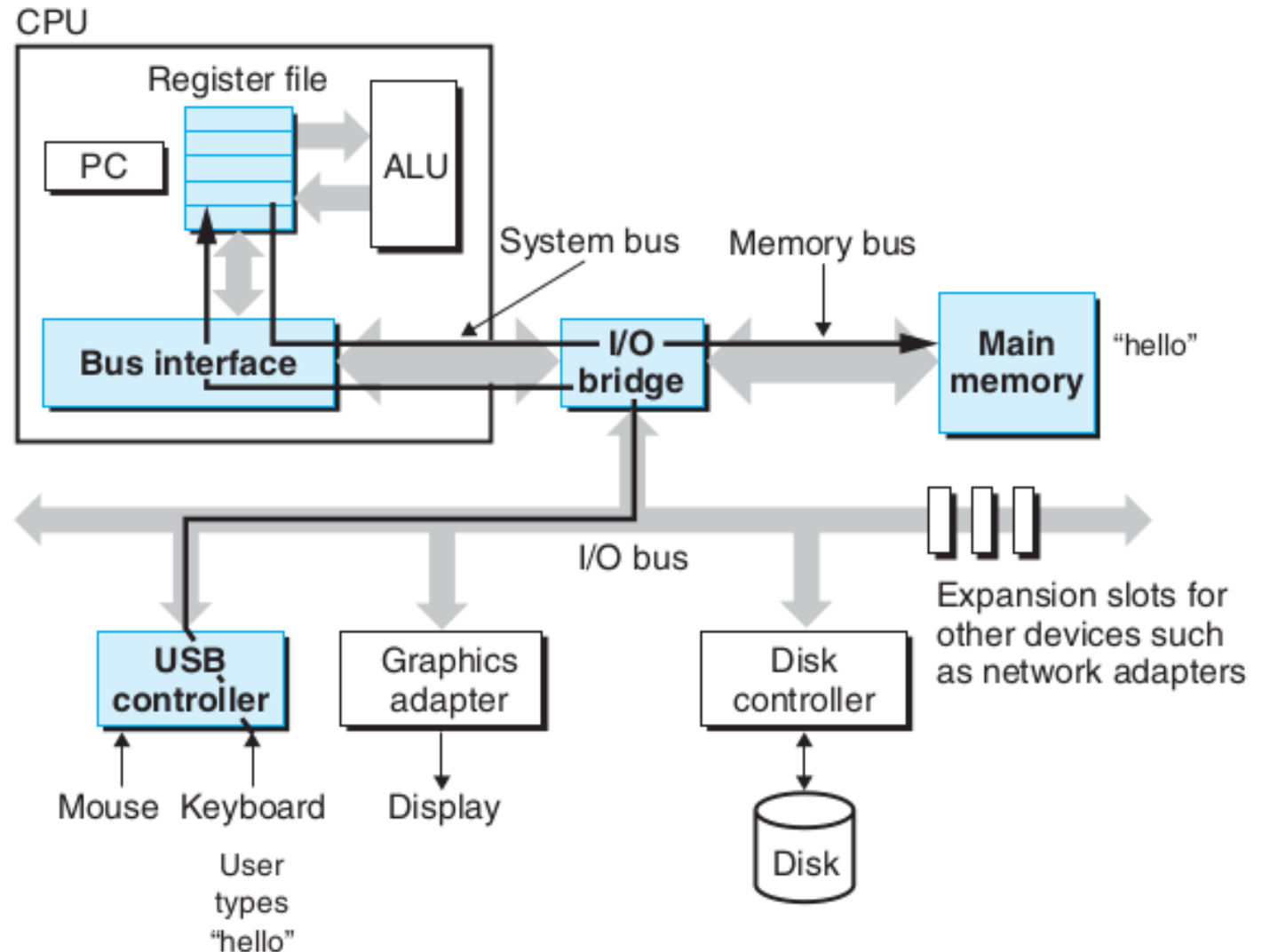
- bit
- byte: 8 bits
- word: usually 4 bytes (32 bits) or 8 bytes (64 bits)

问题：为什么计算机要用0/1位来表示信息？

# 让计算机“动起来”：读取用户输入

Figure 1.5

Reading the hello command from the keyboard.



# 什么是 *shell*?

```
> ./hello  
> ...
```

# 让计算机“动起来”：加载可执行文件

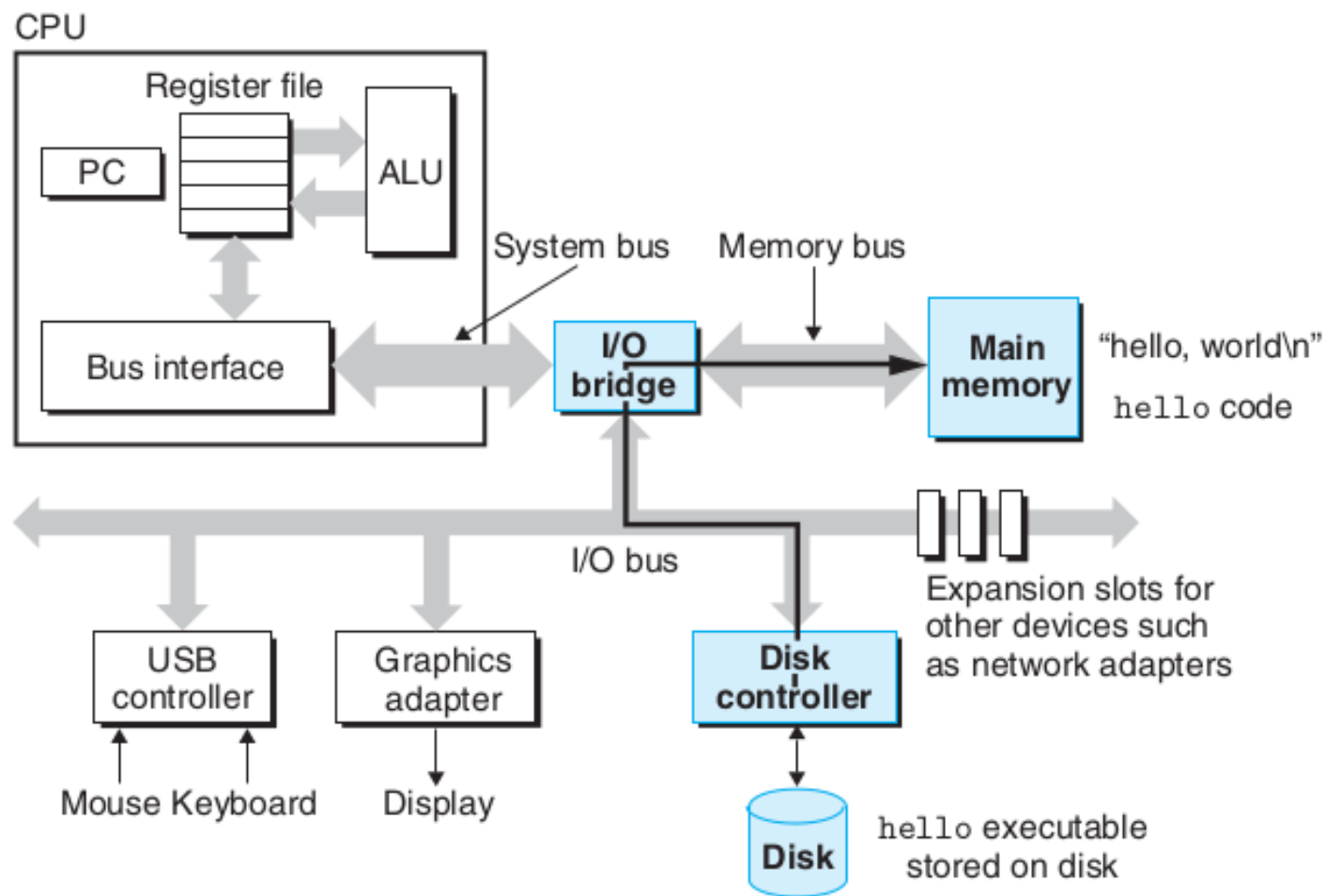
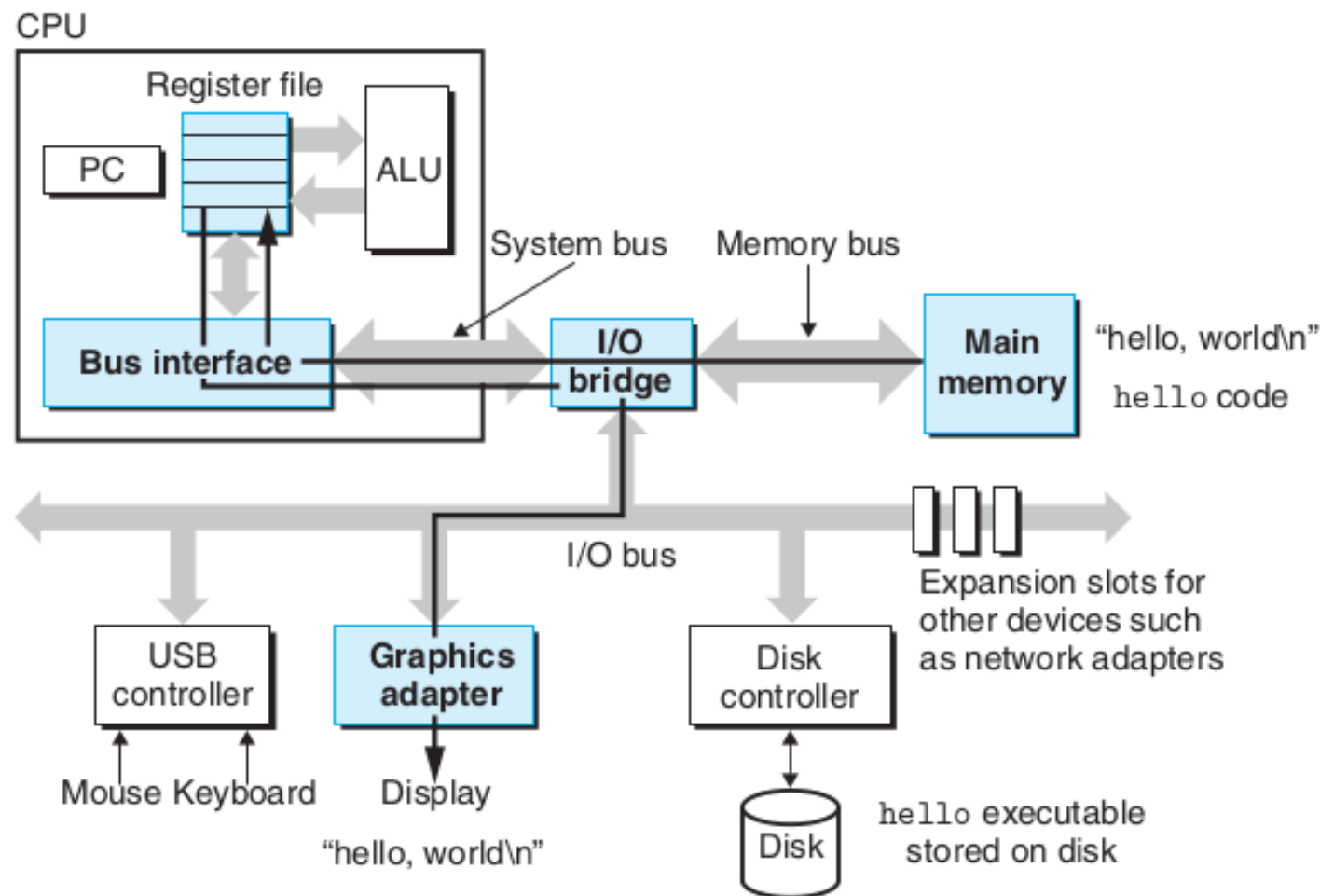


Figure 1.6 Loading the executable from disk into main memory.

# 让计算机“动起来”：输出运行结果

Figure 1.7

Writing the output string from memory to the display.



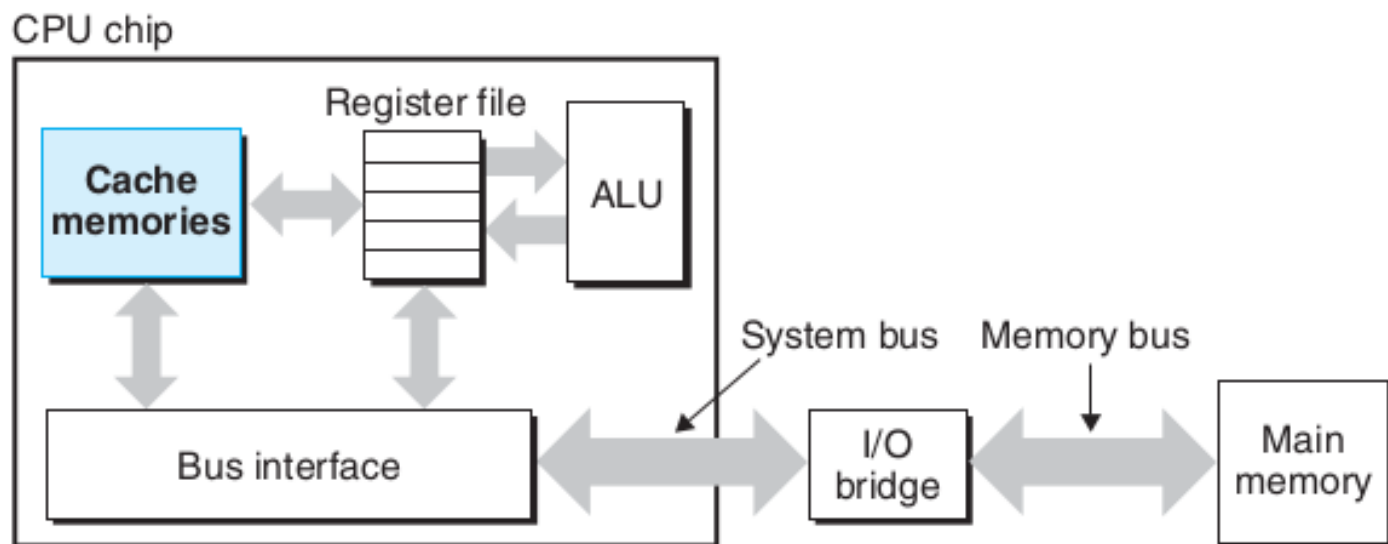
# 计算机系统中的两个重要问题

分层（缓存和存储金字塔）

系统抽象

# 缓存

Figure 1.8  
Cache memories.



# 存储金字塔

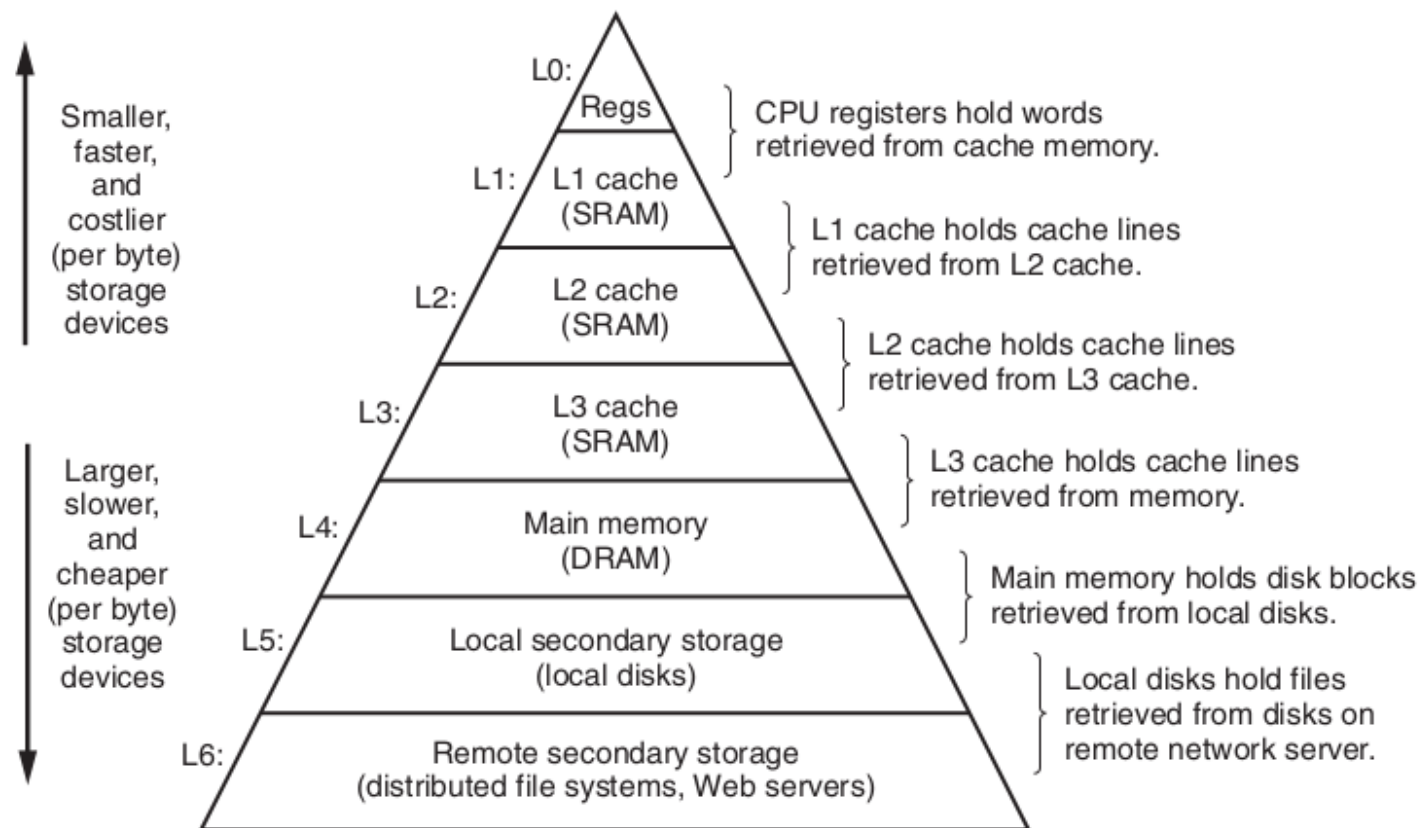


Figure 1.9 An example of a memory hierarchy.



# 计算机系统概览

Figure 1.10  
Layered view of a  
computer system.

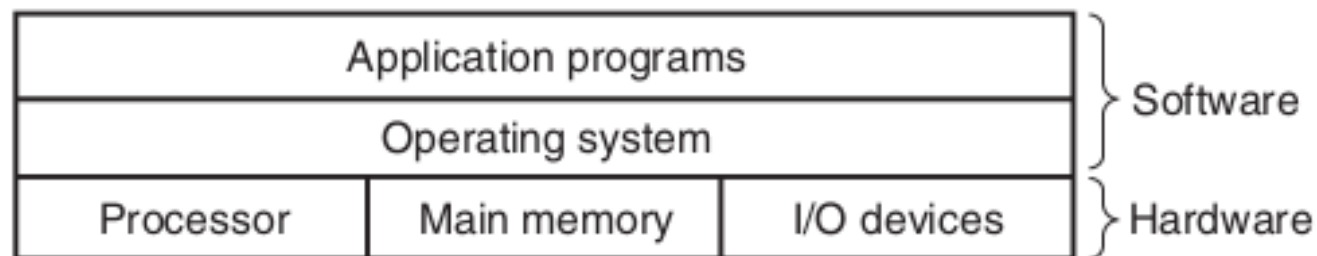
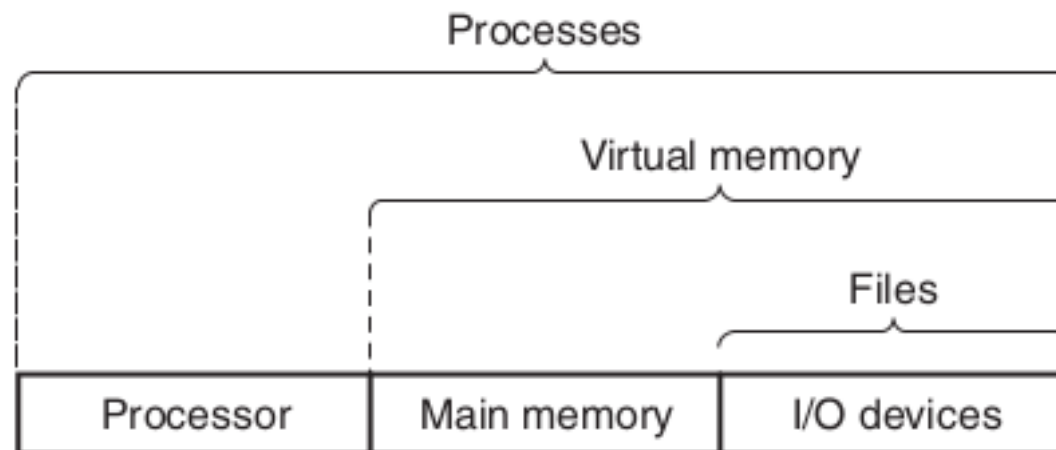
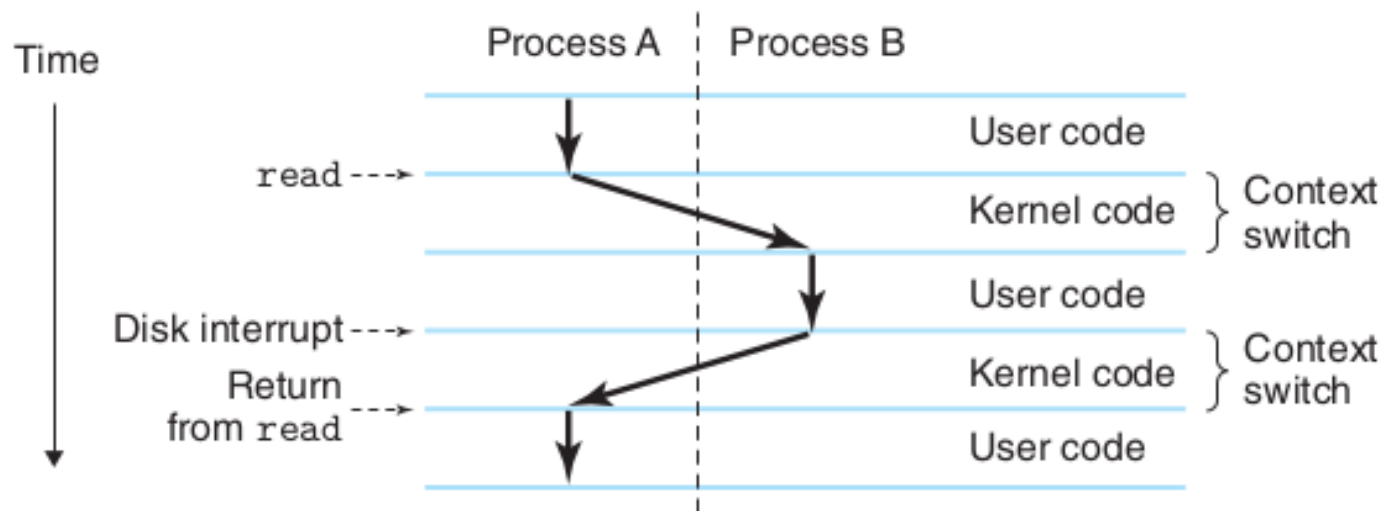


Figure 1.11  
Abstractions provided by  
an operating system.



# 多道程序 (multi-programming)

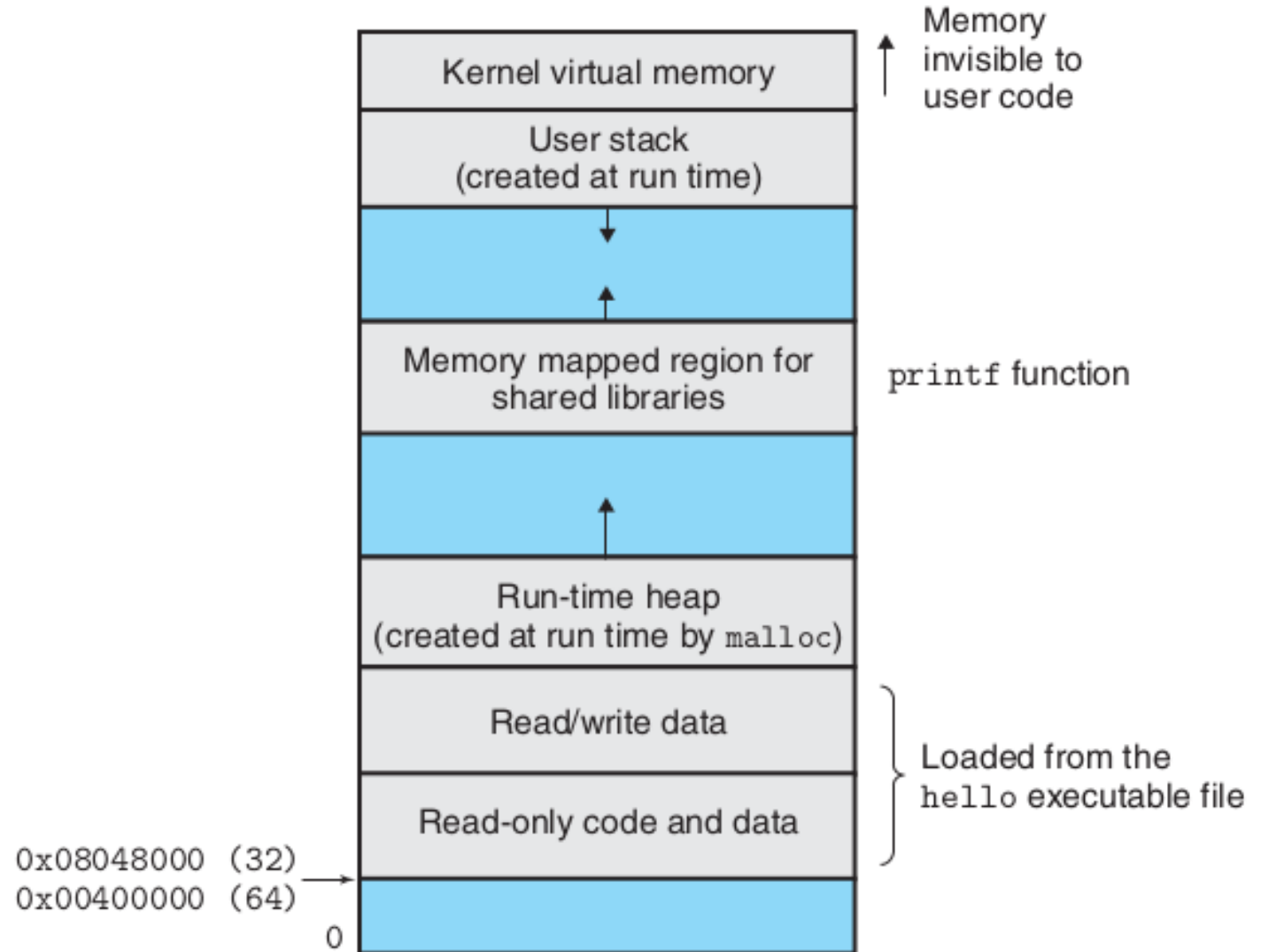
Figure 1.12  
Process context switching.



# 地址空间 (address space)

Figure 1.13

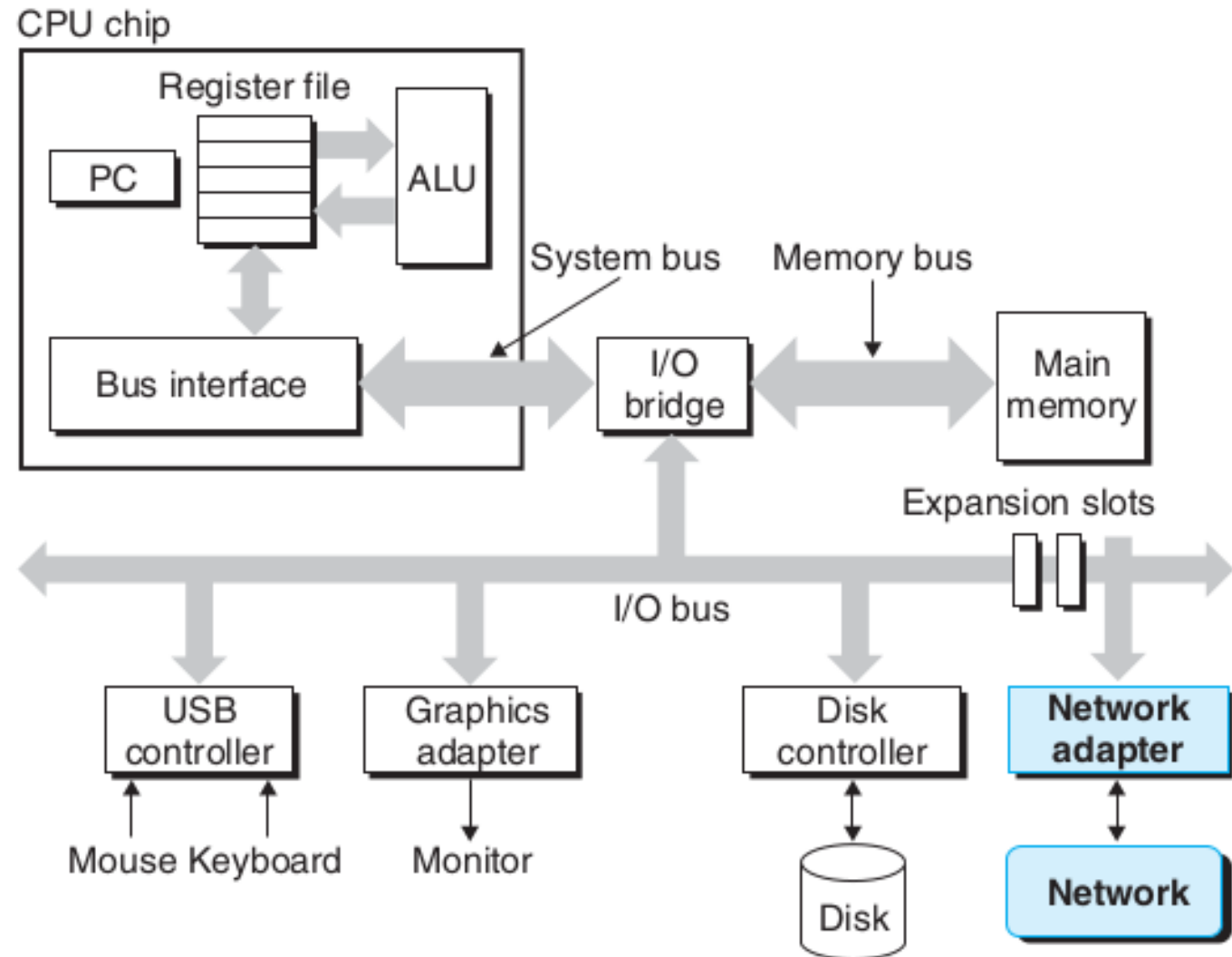
Process virtual address space.



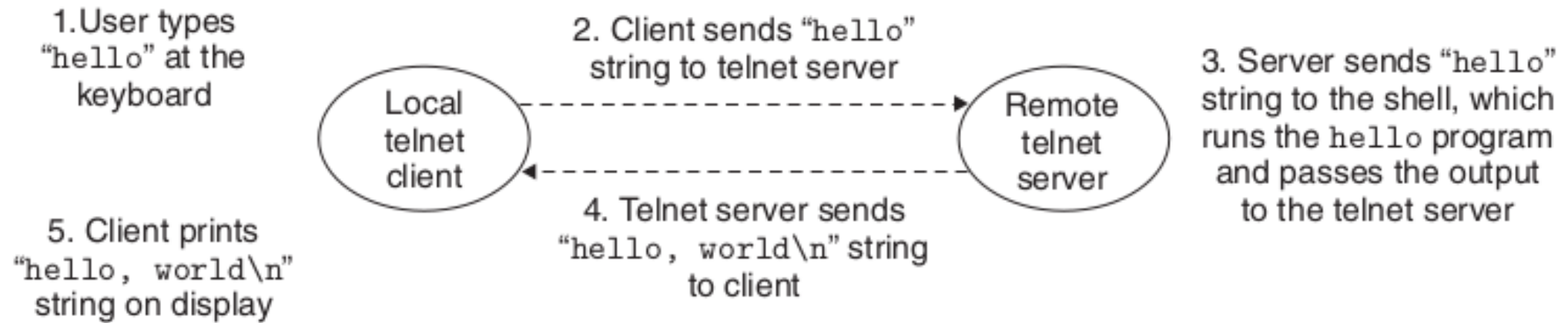
```
$ objdump -d hello  
$ cd /proc/  
$ cat cpuinfo  
$ cat meminfo  
$ top  
$ ...
```

# 网络通讯

**Figure 1.14**  
A network is another I/O device.



# 网络通讯



**Figure 1.15** Using telnet to run `hello` remotely over a network.

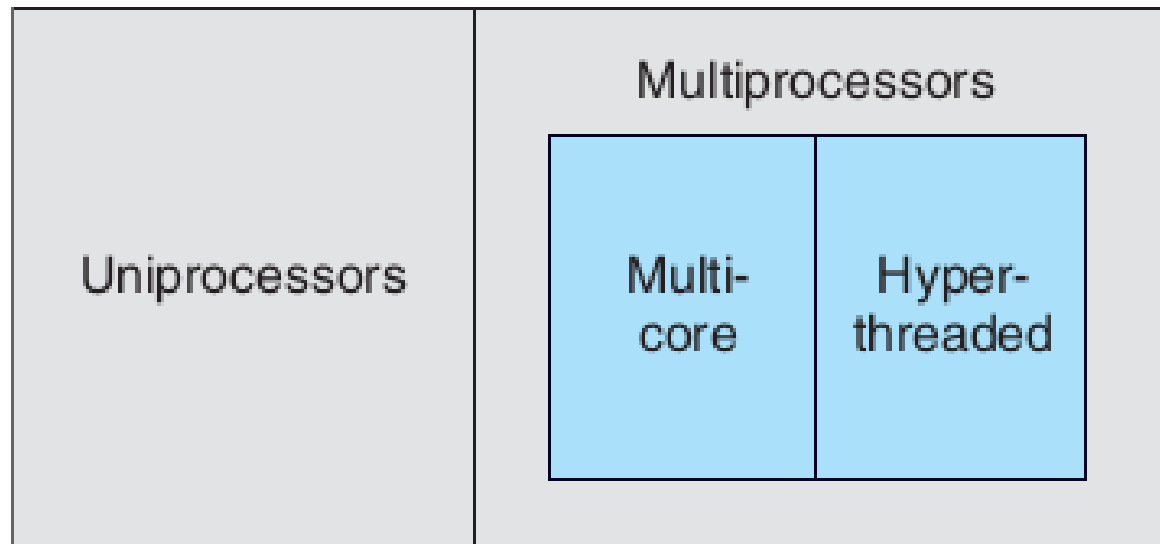
# 单处理器与多处理器

Figure 1.16

**Categorizing different processor configurations.**

Multiprocessors are becoming prevalent with the advent of multi-core processors and hyperthreading.

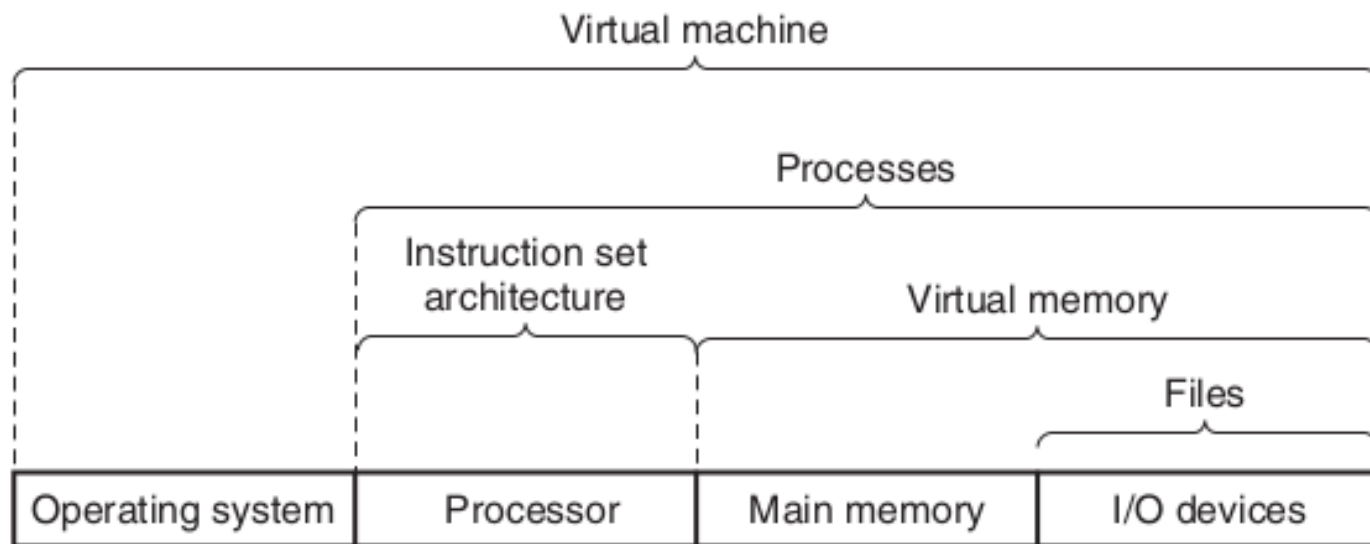
All processors



# 计算机系统提供的抽象

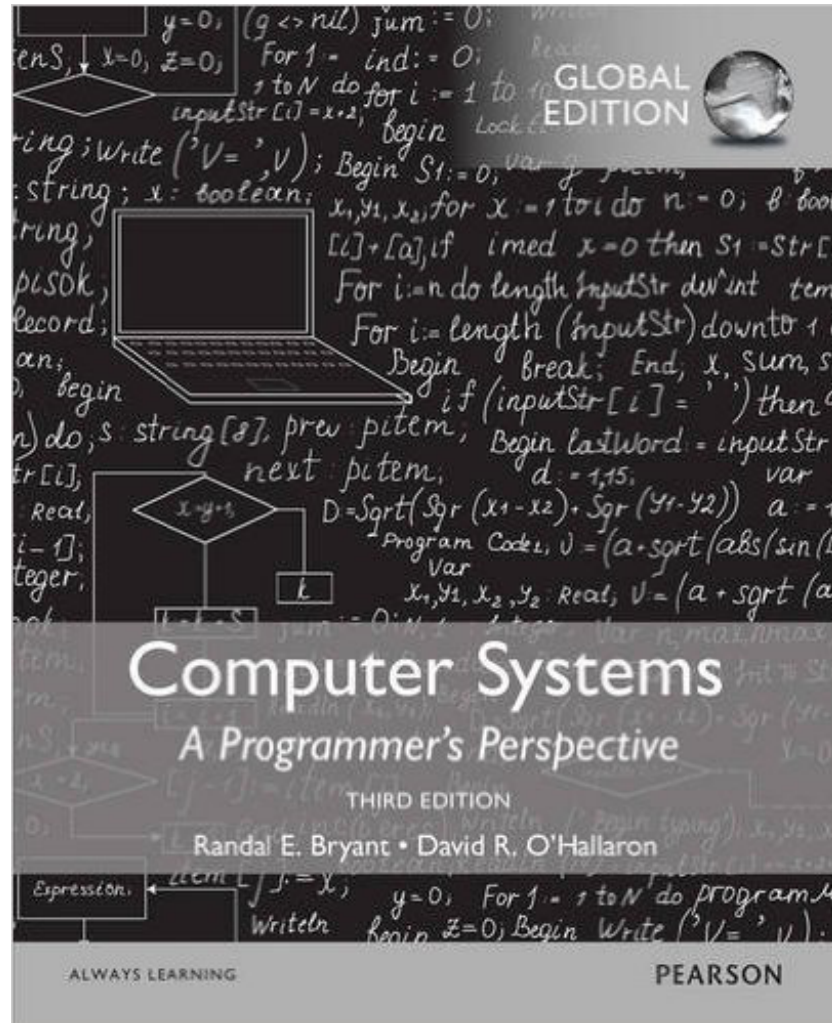
Figure 1.18

**Some abstractions provided by a computer system.** A major theme in computer systems is to provide abstract representations at different levels to hide the complexity of the actual implementations.





# 教材

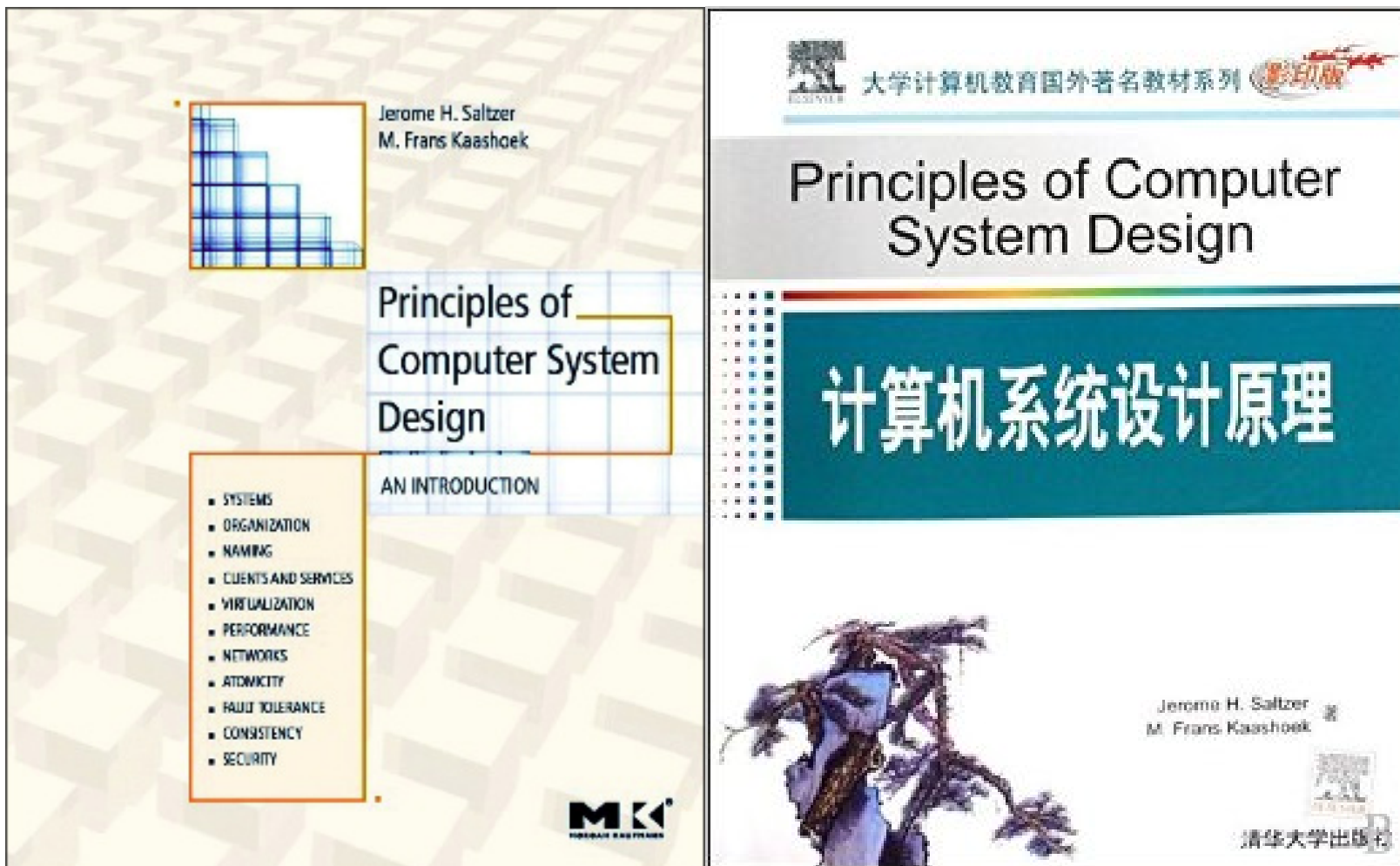


# 课程主要教学内容

Chapter	Topic	Course				
		ORG	ORG+	ICS	ICS+	SP
1	Tour of systems	•	•	•	•	•
2	Data representation	•	•	•	•	⊖ <sup>(d)</sup>
3	Machine language	•	•	•	•	•
4	Processor architecture	•	•			
5	Code optimization		•	•	•	
6	Memory hierarchy	⊖ <sup>(a)</sup>	•	•	•	⊖ <sup>(a)</sup>
7	Linking			⊖ <sup>(c)</sup>	⊖ <sup>(c)</sup>	•
8	Exceptional control flow			•	•	•
9	Virtual memory	⊖ <sup>(b)</sup>	•	•	•	•
10	System-level I/O				•	•
11	Network programming				•	•
12	Concurrent programming				•	•

**Figure 2** Five systems courses based on the CS:APP book. Notes: (a) Hardware only, (b) No dynamic storage allocation, (c) No dynamic linking, (d) No floating point. ICS+ is the 15-213 course from Carnegie Mellon.

# 参考资料



# 要求

## 了解本质

- 第一步：摆脱 GUI 思维（Windows 思维）

## 观察，动手，动脑

- 读代码，读手册，写代码

## 从书本开始，超越书本

- 不是“文科”，不是“PPT课程”

# 要求

- 我们会尝试使用水杉平台 (<http://www.shuishan.net.cn>)
- 安静上课，积极参与
- 独立完成作业和上机任务，保持诚实

# 预习要求

阅读：2.1-2.2.2

抽时间仔细/反复阅读第一章