No.

Date / /

表示文件或目录位置的两种方式 —绝对路径、相对路径

1、绝对路径：从文件系统根目录开始，到目标文件或目录的完整路径
不受当前工作目录影响 (以驱动器字母打头)

2、相对路径：相对某个基准目录来指定位置
.(单点)：表示当前目录 ..(双点)：表示父目录

外存 ①硬磁盘
磁盘读写动作：磁头定位(磁道)、扇区定位和实际读写
划分"逻辑盘"；区分磁电阻效应
②软盘 ③磁带 查找慢，按顺序读写
④光盘 只读、铝膜凹坑与激光原理
⑤U盘(闪存卡) 闪存+USB接口技术

非易失性

输入与输出 设备利用自身的物理电路特性 二进制信息 (与)实际信息
麦克风、音箱要与声卡配合使用

主存与CPU
传输过程 地址总线：CPU把存储单元地址写入其中
CPU向控制总线发出"写"、"读"的信号
数据总线：CPU从中获得或输出数据

ord字转码 chr码转字

地址总线的线数n：存储容量 2^n 个字节 需要n根线 与通常等于宽度

No.
Date / /

电子计算机

第一代 核心技术 电子管 第二代 晶体管 第三代 集成电路

第四代 大规模集成电路 第五代：超大规模

总线：①CPU字长 CPU一次能处理的数据量大小，决定了处理操作数宽度

②数据总线宽度 数据总线一次能传输的最大数据量

影响CPU与内存或其外部设备的数据交换的速度

CPU（微处理器）处理数据的基本单位是字，字长与CPU型号有关，字长以位为单位

为最大化性能角度，数据总线宽度与CPU字长一致，最有效利用每次内存访问

③地址总线宽度 CPU可以寻址的地址空间大小 $2^n B$

④控制总线 传递控制信号，协调CPU与其它组件间操作

不同编码方式下字符占用的字节

编码方式	英文字符	汉字
ASCII	1字节	不支持
GBK	1字节	2字节
UTF-8	1字节	3字节
UTF-16	2字节	常用字2，扩展字4
UTF-32	4字节	4字节

二进制的加减乘除 加：①满2进1

减：①对减数求补码 再相加

乘：整数乘法

#coding:<encoding>

源代码文件开头的注释

声明本文件采用的编码方式。

No.
Date / /

存储系统 (分层存储)

(一)存储系统的层次结构 速度最快,容量最小 → 速度较慢,容量更大

寄存器 高速缓存 主存 外部存储器 远程存储

(二)存储管理技术

① 分层存储 数据根据访问频率存储在不同存储设备 较个速个

② 虚拟内存技术,自动快存储不足

(三)存储管理

㊀硬件 数据物理存储的传输 ① 存储控制器 ⇒ 负责接收读写命令 逻辑地址 ⇒ 物理地址

② 缓存管理 局部性原理

③ 纠错技术

㊁操作系统 ① 虚拟内存技术 负责实现逻辑地址到物理地址

② 分页和分段

③ 文件系统 管理磁盘数据 (本质心)

文件:外存上一组相关数据的逻辑集合

① 存储信息的基本(逻辑)单位

② 有名称属性

I. 文件组织 文件以树形结构组织,只在目录中 每个文件和目录有一个唯一路径

II. 文件元数据 文件系统存储文件的元信息,映射存储块

III. 存储块分配 IV. 日志和事务

文件的本质:字节序列(按顺序排列的字节流)

分类 { 文本文件:字符
二进制文件:应用程序、图形、声音
特定应用于文件相关

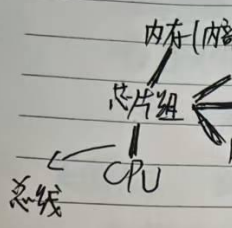
文件系统:操作系统用于管理文件的结构和规则集合

组成 { 文件结构和组织
负责管理文件的软件系统

方式:用目录的分层树状结构将文件组织起来

☆通过文件和文件系统来组织和管理存储在外存储设备上的信息。
文件系统是由文件和文件夹组成的树状结构，对数据进行管理

文件内部数据内容的组织 {顺序存储
数据库：在计算机内建立的一组相关的数据集合
反映的数据管理系统（如Excel



适配器 通过 主板上的插槽 与主板相连
外围设备 通过 适配器上的端口 与适配器相连

显示器与显卡共同完成计算机内信息在屏幕上的显示

互联网三种经典应用
①万维网 网页+超文本链接 格式 HTTP 统一资源定位符 URL
②电子邮件 SMTP POP3
③文件传输 FTP

Python 3.x 的字符串类型 str 是基于 Unicode 编码，
这意味着每个字符（包括汉字）都被视作一个独立字符，
通过对应的 Unicode 码点表示，而不管它在内存中占用多少字节组成

No.
Date / /

数据存储

(一)存储整数　定点表示法

① 无符号表示法　最大无符号整数 2^n-1　(n为计算机中为配置用于表示整数的位数)

② 二进制补码表示法　有符号整数　-2^{n-1}, $2^{n-1}-1$

还原/表示　最左位是1，为负，取补码；是0，不操作

~~③原码表示法~~

(二)存储实数　①定点表示法

②浮点表示法　维持精度　小数点左右有不同数量的数码

符号　位移量　定点数　科学计数法 $+7.4375\times10^{21}$

符号、指数、尾数 ⇒ 无符号表示 → $+7.4375$ 21　$+7.4375E21$

规范化　$+26\times1.00111011$

→ $+$ 6 00111011

只存储小数点后的，1默认

2^n次可

1. continue 跳出当前迭代，继续下一次循环　对定义单个元素的减 (a,)

break 终止整个循环，跳出当前层次的循环

2. Python中运算符的优先级　算术运算 → 位运算 → 比较运算 → 逻辑运算 → 条件运算 → 赋值运算

① 括号 ( ) [ ] { }

② 算术运算　指数 ** ⇒ 正负号 +x, -x, 按位取反 ~x ⇒ 乘除(乘、整) ⇒ 加减

③ 位运算　位移 ⇒ 与 ⇒ 异或 ⇒ 或

④ 比较运算符　>=!

⑤ 身份运算 (is (not)), 成员运算 (not in)

⑥ 逻辑运算 not > and > or

⑦ 条件运算符 if else　⑧ 赋值

No.

Date / /

源码、反码和补码 表示有符号整数的三种常用编码方式

① 源码 将数值的符号和大小分开，使用最高位（最左边的位）表示符号

0正 1负 有两个表示零的方式

正数与原码相同

② 反码 负数的表示是通过取源码每一位的反码

先取正数源码，再对每一位取反 两个0

③ 补码 先表示正数二进制，再对其取反，加再1

0有唯一表示 0000 0000 2次补码返回源码

计算机网络协议 通信双方为实现数据交换所必须遵守的规则、约定

1. 基本概念 语法、语义、时序

2. 网络协议分层模型

OSI七层模型 TCP/IP四层模型

- 网络接口层 = 物理层、数据链路层
- 网络层
- 传输层
- 应用层 = 会话层、表示层、应用层

3. 常见协议

① 网络层协议 • IP 定义数据包的路由规则 IPv4 32位 IPv6 128位

• ARP IP地址转换为MAC 地址类似交互

② 传输层 • TCP 连接、传输可靠 • UDP 无连接、不保可靠、实时通信

③ 应用层协议 • HTTP/HTTPS 超文本传输，HTTPS加密

• FTP 文件传输 • SMTP 电子邮件发送

• DNS 将域名解析为IP地址 (IPv6)

※ IP地址和MAC地址全球唯一，每台计算机必须要有

MAC地址 由网卡决定的唯一性标识（烧录在硬件中）

(OSI模型) 在数据链路层使用，不依赖IP地址，硬件级别标识

※ 局域网内部设备识别、通信

内存(字节数)
字长(字节数)

存储空间 N个字 → 需要 $\log_2 N$ 位来标识每一个字

内存中唯一可标识的单元总数称为地址空间

地址空间　内存中所有可以访问的地址集合范围

地址 一对一 内存中一个存储单元

大小由编址(地址总线宽度)决定　n位 → 2^n 个存储单元

分类：物理地址空间：对应硬件实际内存地址

虚拟地址空间：虚拟内存管理技术，可大于实际物理

存储单元：1字节(8比特) → 最小的寻址单位

数据以二进制形式在内存中传入传出

虚拟地址空间　操作系统管理内存的机制

↓页表

物理　为每个进程提供独立的、隔离的地址空间

通过内存管理单元映射到物理内存，高效利用内存并进程隔离

互联网协议　规定数据在网络中如何寻址、传输、路由

(一)　要素：数据传输(分割为数据包)、寻址(A→B)

路由：(最优路径传递)

① IP协议　数据包寻址、路由　IPv4 → IPv6　地址很多

32位　128位

② TCP协议　数据不丢失、不重复　③ UDP协议　无连接直接发送

④ HTTP　HTTPS　加密　基于文本，在客户端浏览器和服务器传输数据

⑤ DNS　域名 → IP地址

(二) IP地址　唯一标识网络设备　公有、私有、环回

(三) VPN 虚拟专用网络

特点：{临时性：可直接访问任何位置，不需像磁带顺序存储
快速读写：以恒定时间快速地读取或写入

CPU、主存、输入/输出设备

计算机系统 硬件+软件

①硬件系统
- 主存：存储单元集合，每一个单元有个称为地址的标识符
  - RAM 随机存取存储器
  - ROM 只读存储器
- 中央处理器（CPU）=运算器+控制器+寄存器
- 存储设备：1.内存（RAM）：临时存储，断电丢失；2.外存（硬盘，SSD）：长期存储
- 输入设备、输出设备
- 总线：数据总线、地址总线、控制总线

②软件系统
- 操作系统（OS）管理硬件资源，协调各软件、硬件自己工作，任务调度
- 应用程序
- 开发工具

特殊寄存器：程序计数器（PC）

快速独立的存储设备

(二) 计算机使用 能为程序员的一组指令来处理数据

取指令 → 译码 → 执行 → 写回 → 机器周期

计算机分类 CISC（复杂指令集计算） RISC（精简指令集）

使用流水线技术：控制单元同时执行 2~3 个所指

存储每个像素的位模式

图像、视频文件的内存

(一) 图象：存储大小（字节）= 分辨率高度 × 分辨率宽度 × 色深（位）/ 8

光栅图（位图）、矢量图

RGB 模型 8位色深 3字节

分辨率 × 每个像素的字节数（色深）

(二) 视频：存储大小（字节）= 分辨率高 × 宽 × 色深（位）/ 8 × 帧率（FPS）× 时长（s）

每帧位图

比特率：每秒钟传输的数据量 Mbps

动画为矢量图

存储大小（字节）= 比特率（bps）÷ 8 × 时长（s）

(三) 声音：存储大小（字节）= 采样率（Hz）× 每个采样点的位数（位深）/ 8 × 声道数 × 时长（秒）

模拟信号 → 量化

No.
Date / /

CPU访问程序的性质

① 顺序执行性　程序指令顺序执行　控制流可改变执行路径

② 指令周期　取指令解码.执行　指令：组成程序的基本单位

③ 内存访问的局部性　1.时间局部性　~~某位置被多次访问~~

程序中某些指令或数据会在短时间内反复访问

2.空间局部性　若访问某地址附近一个位置，则可能在不久将来访问该位置附近

④ 指令流水线　多个指令可分别执行不同阶段（如指令流）

⑤ 并行性　多核心处理

⑥ 缓存机制　⑦ 多级内存层次结构　CPU寄存器→缓存→主存→硬盘

⑧ 输入.输出设备和程序交互

CPU通过I/O通道或直接存储器访问

缓存　由于内存访问速度慢，CPU引入缓存，

缓存机制使CPU经常访问的程序和数据存储更接近CPU的位置　CPU　L1缓存　L2缓存　主存　L3缓存（多核共享）

操作系统（OS）　功能　① 资源管理　② 提供用户接口　③ 提供运行环境

连接硬件　CPU管理.内存管理.设备管理.文件管理

于应用软件的中间层软件　操作系统的引导：将其内核从磁盘装入主存，并使它正常工作

CPU的地址总线宽度决定它能够直接寻址的内存空间大小

每一根地址线对应两个状态（0/1），每个地址是一个字节（Byte）

总线宽度（位）n ⇒ 总内存空间　2^n B

在一般程序中，变量和存储单元相对应　变量{名字 内容} ⇒ 地址 数据} 存储单元

10^3 10^6 10^9 10^{12}
K M G T

No.
Date / /

1. CPU计算速度公式
CPU计算速度 = 时钟频率 × 每周期指令数 × 核数
GIPS GHz IPC

2. 计算机发展史的重要人物
① 查尔斯·巴贝奇 计算机之父 存储器、输入输出设备、中央处理器 差分机
② 阿达·洛夫莱斯 提出程序 | 第一代通用完全电子的计算机
③ 艾伦·图灵 ④ 冯·诺依曼 ⑤ 香农 信息论 约翰·莫奇勒
芯片速度每18个月增加倍 ⑥ 戈登·摩尔 摩尔定理 ⑦ 丹尼斯·里奇 C语言 Unix操作系统
⑧ 比尔·盖茨 windows ⑨ 乔布斯
⑩ 蒂姆·伯纳斯-李 万维网 www, HTTP
⑪ 林纳斯·托瓦兹 Linux操作系统

3. 2个字节比特表示整数的范围 (补码)
8bit → 16位二进制 − 1位符号 = 15位 范围 -2^{15-1} 到 $2^{15-1}-1$
-32768 32767

4. 计算机用二进制表示数据时，小数可能由于位数限制不能精确表示

5. 计算机机箱
① 主板：所有硬件的核心连接平台 适配器：插卡，连接主板和外部设备（显卡、网卡、无线网卡）
② 中央处理器(CPU) 执行计算、逻辑操作
③ 内存 ④ 存储设备 ⑤ GPU 处理图形、图像数据 → 显示 输入输出设备

6. IP地址的点分十进制 8位一组，二转十，10点
域名：互联网上网站的别称，易于记忆，字符串存在，可通过DNS与IP关联

IPv6 十六进制冒号分隔法
2^{128} 128位 每16位作为进制的4位，冒号分

No.
Date / /

图灵机 抽象计算模型，描述计算过程，模拟计算机程序执行

(一)原理 (存储数据) (移动、操作) (当前、初始、终止)

无限长纸带(Tape) 读写头(Head) 状态(State)

状态转移的函数 接受和拒绝状态

(决定机器行为) (成功 or 失败)

(二)工作过程 初始状态，状态转移，重复执行，进入终止 (接受/拒绝)

形式化定义：七元组 $M=(Q, \Sigma, \Gamma, \delta, q_0, q_{acc}, q_{re})$

Q：状态的有限集合；Σ：输入符号的集合；Γ：纸带符号集合；q_0：初始状态；δ：状态转移函数

(三)停机问题：无法判断某些程序是会终止

无法解决

$\delta: Q \times \Gamma \to Q \times \Gamma \times \{L, R\}$

(当前Q，指至Q，左右移动)

冯诺依曼结构

1、部分 运算器 (ALU) 处理数据，运算 (算术逻辑单元)

(控制单元)控制器 (CU) 指挥协调计算机各部件 从内存中读取，解释指令

存储器 (Memory)：保存程序指令和数据

重要特点：统一存储

程序、数据用同样存放

输入设备，输出设备

2、特点：①统一存储，二进制编码 ②按指令顺序执行

③单一通信通道 一条总线连接CPU与内存

No.

Date / /

位运算：对整数的二进制位操作的运算

(部分)复位置0 掩码与

① 按位与 (and) → &
两个二进制位都为1，才为1，否则为0

② 按位或 (or) → |
两个二进制位有个为1，则为1

(部分)指定位反转 掩码异或(为1反转，为0不变)

③ 按位异或 (XOR) → ^
两个二进制位不同结果为1

④ 按位取反 (NOT) → ~ 求补
将每一位反转

⑤ 左移：→ <<
补0 相当于乘以2的n次方

⑥ 右移 → >>
补0

Python 标识符规则

1. 只包含字母 (A-Z, a-z)，数字 (0-9) 和下划线 (_)
2. 不能以数字开头

字节 Byte 单位

比特 (bit) 计算机存储的最小单位，表二进制的一个位 (0或1)

比特 (bit)	位	0或1 代表某个状态
字节 (B)	1B=8bit	
千字节 (KB)	1KB=1024B	$2^{10}B$
兆字节 (MB)	1MB=1024KB	$2^{20}B$
千兆字节 (GB)	1GB=1024MB	$2^{30}B$
太字节 (TB)	1TB=1024GB	$2^{40}B$

进制转换

十进制 → 其他

短除法取余 (整数)

小数部分乘以目标进制，取出整数部分作为当前位的值

其他 → 十进制

权重展开法

$\square\square\square.\square\square$

$K^2\ K^1\ K^0 . K^{-1}\ K^{-2}$

178.6

178 短除取余 向左写

0.6 乘进制取整 向右写

二进制转 8、16 (3位一组；4位一组)