计算机基础知识与数据结构知识点整理

1. 一、计算机基础知识

1.1 （一）著名人物及成就

姓名	国籍	主要贡献	称号（身份）
艾伦·麦席森·图灵	英国	图灵机模型、图灵测试	计算机科学之父、人工智能之父
约翰·冯·诺依曼	美籍匈牙利人	冯·诺依曼架构、提出存储程序	计算机之父、博弈论之父
克劳德·艾尔伍德·香农	美国	提出信息熵、通讯复杂性	信息论创始人
姚期智	中国	伪随机数生成	2000 年图灵奖，唯一华裔获奖学者
戈登·摩尔	美国	提出摩尔定律	Intel 创始人之一
阿达·洛芙莱斯（女）	英国	建立循环和子程序概念	计算机程序创始人、世界第一位写程序的人
董铁宝	中国	中国计算机研制和断裂力学研究的先驱之一	中国计算机之父、中国第一位写程序的人

1.2 （二）计算机应用方向

应用方向	涉及领域	具体示例
科学计算（数值计算）	天气预报、数学及物理研究等需大规模计算的领域	气象数据模拟、物理方程求解
信息处理	需对各种信息做采集、存储、分类、统计、加工、传递和使用的领域	企业客户信息管理、学生成绩统计
过程控制	工业、农业、交通运输等领域中用于实时监测和控制各种生产过程	工厂生产线温度控制、农业大棚湿度调节、交通信号灯控制
辅助设计与制造	CAD（计算机辅助设计）、CAM（计算机辅助制造）等相关领域	汽车零部件设计、机械加工流程规划
网络通信	各类依赖网络传输数据的领域	电子商务（网上购物）、即时通讯（微信聊天）、远程教育（在线课程学习）
人工智能	涉及机器模拟人类智能的领域	机器视觉（图像识别）、自然语言处理（语音助手）、大模型（ChatGPT 类应用）

1.3 （三）国内赛事

1. **背景**：1984 年邓小平指出 “计算机的普及要从娃娃做起”，中国计算机学会（CCF）于 1984 年创办全国青少年计算机程序设计竞赛（简称：NOI）。
2. NOI 系列活动
   - 全国青少年信息学奥林匹克联赛（NOIP）
   - 全国青少年信息学奥林匹克竞赛（NOI）
   - 全国青少年信息学奥林匹克竞赛冬令营（WC）
   - 国际信息学奥林匹克中国队选拔（CTS）
3. **国际赛事**：进入国家队的选手将参加国际信息学奥林匹克竞赛（IOI）。

1.4 （四）硬件系统

1. 中央处理器（CPU）
   - 地位：计算机系统的运算和控制核心，是信息处理、程序运行的最终执行单元。
   - 组成部分及功能
     - 运算器：进行各种算术运算（如加、减、乘、除）和逻辑运算（如与、或、非）。
     - 控制器：计算机的指挥控制系统，负责协调计算机各部件有序工作。
     - 寄存器：用于暂时存储指令、地址等数据，是 CPU 的组成部分。
2. 输出设备
   - 定义：将计算机的各种结果数据或信息以数字、字符、图像、声音等形式表现出来的设备。
   - 常见设备：显示器、打印机、绘图仪、音箱等。
3. 总线（BUS）
   - 定义：计算机系统各部件之间相互连接、传输信息的公共通道。
   - 分类及功能
     - 数据总线（DB）：双向总线，具体方向由 CPU 控制，用来传送数据信息，主要链接 CPU 与各个部件，是它们之间交换信息的通路。
     - 地址总线（AB）：单向总线，用来传送 CPU 向外发出的地址信息，地址总线的宽度决定可以访问的存储器容量大小，如 20 条地址总线可以控制 1MB 的存储空间（2^{20}B=1MB）。
     - 控制总线（CB）：双向总线，由具体控制信号而定，用来传送控制信号，以协调各部件之间的操作，控制信号包括 CPU 对内存储器和接口电路的读写控制信号、中断响应信号，也包括其它部件传送给 CPU 的信号，如中断申请信号、准备就绪信号等。
4. 存储器相关
   - 内存储器存储单元：每个存储单元都被赋予一个唯一的序号，称为地址。
   - 易失性与非易失性存储器：断电后会丢失数据的存储器是 RAM（随机存取存储器），ROM（只读存储器）、硬盘、光盘断电后数据不会丢失。
   - 寄存器归属：寄存器是中央处理器（CPU）的重要组成部分。
   - CPU 访问最大存储器容量决定因素：取决于地址总线。

1.5 （五）软件系统

1. **系统软件分类及示例**

类别	具体软件及说明
操作系统	Unix（现代操作系统的基石）、Linux（免费使用和传播的类 Unix 系统）、Windows（软件开发，窗口式操作系统）、Mac OS（苹果公司开发，运行于 Mac 电脑）、鸿蒙（华为公司开发，主要面向物联网）
语言处理程序	汇编器（将汇编语言翻译为机器可以执行的代码）、编译器（将高级编程语言书写的源程序，翻译成等价的机器代码）、链接器（将一个或多个由编译器或汇编器生成的目标文件及所需库文件链接为一个可执行文件）、解释器（能够把高级编程语言一行一行直接转译运行，无需在程序执行前将整个源代码编译成机器码，如 Python、JavaScript 等）
数据库管理系统	MySQL、PostgreSQL、SQL Server、SQLite、MongoDB、Oracle 等
其他支持软件	各类为系统运行提供辅助支持的软件

2. 常见软件相关考题要点

- 非操作系统：Notepad（记事本，文本编辑软件）不是操作系统。
- 编译器功能：将源代码转换为机器代码，不直接执行源代码、不进行代码调试、不管理程序运行时的内存。
- 操作系统功能：控制和管理计算机系统的各种硬件和软件资源的使用，不是负责外设与主机之间的信息交换、诊断机器故障、将源程序编译成目标程序。
- 微软公司软件：Powerpoint、Word、Excel 是微软出品，Acrobat Reader 是 Adobe 公司出品。
- 图像格式：JPEG、GIF、PNG 是图像格式，TXT 是文本格式。

1.6 （六）信息编码

1. **计算机数据存储基础**：计算机只认识二进制数，图片、数据、文字、变量、邮件、视频等在计算机中都被存储为 0 和 1 的组合。

2. **不同类型数组内存占用计算（假设元素个数 n 最大为 10000000）**

数据类型	每个元素字节数	内存占用大小（MB）	计算过程
布尔型（bool）	1	≈9.5	1×10000000÷1024÷1024≈9.5
整型（int）	4	≈38.1	4×10000000÷1024÷1024≈38.1
浮点型（double）	8	≈76.3	8×10000000÷1024÷1024≈76.3

3. 存储单位换算

- 1B（字节）=8bit（比特）
- 1KB（千字节）=1024B
- 1MB（兆字节）=1024KB
- 1GB（吉字节）=1024MB
- 1TB（太字节）=1024GB

4. 非数字编码发展阶段

- ASCII 码阶段：最早的编码标准，由美国国家标准学会指定，包括数字、大小写字母及一些符号等。

- 各国编码阶段：各国制定自己的编码标准处理本国语言，如中文的 GB2312、后续扩展的 GBK，以及繁体中文 BIG5 等。
- Unicode 阶段：解决多种语言编码冲突和乱码问题，将所有语言的字符统一到一个编码体系，通常需 2 个字节表示一个字符，大量使用英文时存储和传输效率较低。
- UTF-8 阶段：解决 Unicode 效率问题，是可变长编码，英文字符用 1 个字节，汉语通常用 3 个字节表示。

5. 图像与视频存储容量计算
   - 图像存储容量：分辨率 × 位深 ÷8（单位：字节），如分辨率为 2048×1024 像素的 32 位真彩色图像，存储容量为 2048×1024×32÷8÷1024÷1024=8MB。
   - 视频存储容量：图像存储容量 × 帧率 × 时长（秒），如 8 分钟（480 秒）、24 帧 / 秒、2048×1024 像素、32 位真彩色的视频，存储容量为 2048×1024×32÷8×24×480÷1024÷1024÷1024≈30GB。
6. **颜色编码位数**：表示 256 种颜色，至少需要 8 位二进制（2^8=256）。
7. 格雷码
   - 提出背景：由贝尔实验室工程师法兰克・格雷于 1953 年提出，目的是减少在数字转换过程中可能出现的错误。
   - 基本特点：任意两个相邻的编码只有一位二进制不同（第一个和最后一个也相邻）；编码不能出现重复。
   - 二进制码与格雷码转换规则
     - 二进制码转格雷码：保留二进制码的最高位作为格雷码的最高位（最高位不变）；格雷码的其余位是二进制码当前位与二进制码上一位异或的结果（相同为 0，不同为 1）。
     - 格雷码转二进制码：最高位不变；二进制的其余位是格雷码当前位与二进制码上一位异或的结果。
   - 作用：在数字系统中使用格雷码可以减少相邻状态变化时的错误率。

1.7 （七）计算机网络与安全

1. 网络主要功能
   - 数据通信：依据一定的通信协议，利用数据传输技术在两个终端之间传递信息。
   - 资源共享：包括硬件资源、软件资源和数据资源的共享。
   - 集中管理：通过现代化管理系统实现日常工作的集中管理，提高效率。
   - 分布式处理：通过算法将大型综合性问题交给不同计算机同时处理。
   - 负载均衡：将工作均匀分配给网络上各台计算机系统，某台计算机负荷过重时，自动转移负荷到较轻的计算机系统。
2. **网络分类**

分类依据	类别	特点及示例
地理范围	局域网（LAN）	较小地理区域（约 10 公里内），如公司内部局域网，实现内部资源共享和通信
	城域网（MAN）	城市范围内（约 10~100 公里），用作骨干网，连接同一城市内的主机、数据库及 LAN 等，如宽带城域网
	广域网（WAN）	跨大物理范围（可达几千公里），连接不同地区局域网或域域网，如因特网
传输方式	有线网	采用双绞线、同轴电缆或光纤连接，建设时间长、成本高，但稳定性好
	无线网	无需布线，使用电磁波（微波、红外线、激光等）通信，建设成本低、时间短，但传输范围有限，易受环境影响
拓扑结构	总线型	共享通路物理结构，结构简单、节省电缆、扩充或删除结点容易；但总线任务重、连接结

点不宜过多，总线故障影响整个网络 |

| | 环型 | 网络结点连接成闭合结构，信号单向传输，适用于实时控制局域网；节省电缆（与总线型相当，低于星型），但增加新结点麻烦，单个站点故障影响整个网络 |

| | 树型 | 容易扩充网络规模；但层级越高的结点故障，导致的问题越严重 |

| | 网状型 | 每个结点至少与其他两个结点相连，可靠性高，常用在广域网中；但结构复杂、不易管理维护、费用较高 |

| | 混合型 | 同时使用以上两种或两种以上基本拓扑结构 |

3. 网络体系结构

- OSI 七层模型：从下到上依次为物理层、数据链路层、网络层、传输层、会话层、表示层、应用层。
- TCP/IP 模型：分为四层（从下到上：网络接口层、网络层、传输层、应用层）和五层（从下到上：物理层、数据链路层、网络层、传输层、应用层），实际大多使用 TCP/IP 四层和五层模型。

4. 电子邮件协议

- SMTP 协议（简单邮件传输协议）：用于发送电子邮件，规定发送方和接收方的连接传输规则。
- POP 协议（邮局协议，POP3 即第 3 个版本）：用于从电子邮件服务器检索邮件，允许用户将邮件从服务器下载到本地。
- IMAP 协议（互联网信息访问协议）：从电子邮件服务器检索邮件，提供比 POP 更强大的访问和管理功能。

5. IP 地址

- 分类：Internet 委员会定义 5 种 IP 地址类型，A、B、C 类在全球统一分配，D、E 类为特殊地址。
- A、B、C 类地址区别

  | 类别 | 网络地址与主机地址划分 | 最大网络数 | IP 地址范围 | 私有 IP 地址范围 | 单个网段最大主机数 | 适用场景 |

  | ---- | ---- | ---- | ---- | ---- | ---- | ---- |

  |A 类 | 第 1 个字节是网络地址，后 3 个字节是主机地址 | 126（2^7-2）|1.0.0.1~127.255.255.254|10.0.0.0~10.255.255.255|16777214 | 大型组织或国家级网络 |

  |B 类 | 前 2 个字节是网络地址，后 2 个字节是主机地址 | 16384（2^{14}）|128.0.0.1~191.255.255.254|172.16.0.0~172.31.255.255|65534 | 中型企业或政府机构 |

  |C 类 | 前 3 个字节是网络地址，后 1 个字节是主机地址 | 2097152（2^{21}）|192.0.0.1~223.255.255.254|192.168.0.0~192.168.255.255|254 | 小型企业或校园网 |

- 书写规则：用 “.” 分成 4 段，每段包含 8 位二进制数，对应十进制范围从 0~255。

6. 域名

- 作用：解决 IP 地址不方便记忆且不能显示组织名称和性质的问题，由域名系统 DNS（Domain Name System）统一管理，DNS 的作用是将人类可读的域名转换为机器可读的 IP 地址。
- 顶级域名分类
  - 国家顶级域名：如 cn（中国）、us（美国）、uk（英国）、kr（韩国）、jp（日本）。
  - 国际顶级域名：int，国际性组织可在 int 下注册。
  - 通用顶级域名：com（企业）、edu（教育）、gov（政府）、org（非营利性）。

7. **计算机病毒特性**

| 特性 | 说明 |

| ---- | ---- |

| 寄生性 | 大多依附在别的程序上 |

| 隐蔽性 | 通常悄然无声进入系统，不易察觉 |

| 潜伏性 | 通常潜伏在计算机程序或磁盘中，在一定条件下发作 |

| 传染性 | 能够自我复制，并通过一定媒介传播 |

| 破坏性 | 会占用计算机一定的资源，严重的会破坏数据，甚至造成大面积的计算机瘫痪 |

1.8 （八）编程语言

1. **编程语言发展阶段**

| 代次 | 特点及示例 |

| ---- | ---- |

| 第一代 | 机器语言，用二进制代码表示的计算机能直接识别和执行的指令 |

| 第二代 | 汇编语言，用英文助记符来表示指令，在一定程度上克服了机器语言难读难改的缺点 |

| 第三代 | 高级语言，以人类日常语言为基础，使用人易于接受的文字来表示，编写程序更简单、可读性更高，如 BASIC、Fortran、Pascal、C、C++、C#、Java、Python 等 |

| 第四代 | 面向问题的语言（高生产率语言） |

| 第五代 | 人工智能语言 |

2. **C++ 语言面向对象特性** ：调用 printf 函数不涉及 C++ 面向对象特性，调用用户定义的类成员函数、构造 class 或 struct、构造来源于同一基类的多个派生类均涉及面向对象特性。

2. 二、数据结构 1

2.1 （一）栈

1. 概念

- 数据操作：只从一端完成（如存、取等）。
- 数据存放原则：先进后出、后进先出。
- 关键位置：最先进入的位置叫栈底，最后进入的位置叫栈顶。
- 基本操作：入栈（进栈，数据存入栈中）、出栈（退栈，数据从栈中取出）。

2. **出栈序列合法性判断**：根据入栈顺序和栈 “先进后出” 原则判断，例如入栈顺序为 4、9、5、3、2 时，出栈序列 5、4、3、2、9 为不合法序列（因为 4 入栈后，9 在 4 上方，4 无法在 9 之前出栈）。

2.2 （二）波兰表达式（逆波兰表达式）

1. **逆波兰表达式转换**：将中缀表达式转换为后缀表达式（逆波兰表达式），消除括号，使运算符顺序符合计算优先级，例如 5+(3-2×4) 转换为 5 3 2 4 × - +。

2. **逆波兰表达式运算**：从左往右找到第 1 个运算符，将运算符左侧两个数字计算结果重新放回表达式中，重复此步骤直至得到最终结果，例如计算 5 3 - 2 4 + ×，先计算 5-3=2，表达式变为 2 2 4 + ×，再计算 2+4=6，表达式变为 2 6 ×，最后计算 2×6=12。

3. 表达式转换与计算考题要点

- 后缀表达式转前缀表达式：根据后缀表达式运算顺序反向推导，例如后缀表达式 6 2 3 + - 3 8 2 / + * 2 ^ 3 + 对应的前缀表达式是 + ^ * - 6 + 2 3 + 3 / 8 2 2 3。
- 中缀表达式转前缀 / 后缀表达式：如 a - b + c × d 的前缀表达式是 + - a b × c d；a × (b + c) × d 的后缀表达式是 a b c + × d ×。
- 前缀表达式计算：从右往左按运算符顺序计算，例如前缀表达式 + 3 * 2 + 5 12，先计算 + 5 12=17，再计算 * 2 17=34，最后计算 + 3 34=37。

2.3 （三）队列

1. 概念

   - 数据操作端口：一个端口负责进，另一个端口负责出。
   - 数据存取特点：先进先出。

2. **常用操作及相关概念**

概念	解释
队首	允许删除队列元素的一端
队尾	允许插入队列元素的一端
空队列	队列中没有元素的状态
入队操作	向队列中插入元素的操作
出队操作	从队列中删除元素的操作
队列长度	队列中的元素个数

2.4 （四）链表

1. **基本概念**：把分散在内存中的数据通过指针串联起来的存储结构，链表中的元素（结点）由两部分组成，一部分存放数据，另一部分存放指针。

2. 链表分类

   - 单链表：结点只能访问后继结点，每个结点包含数据域（data）和指针域（next），最后一个结点的指针域为 NULL。
   - 双向链表：在单链表基础上，每个结点添加 prev 指针，prev 指向前一个结点，可访问前驱和后继结点。

3. 链表特点

   - 优点：不必事先估计存储空间，插入和删除操作不需要移动大量元素，所需空间与线性表长度成正比。
   - 缺点：不可随机访问任意元素，内存地址可不连续（部分地址不必连续）。

4. 链表操作考题要点

   - 单链表结点交换：交换两个结点需保持链表连续，注意指针指向顺序，例如交换 p 和 r 所指结点，需先让 p->next 指向 r->next，再让 r->next 指向 q，最后让 q->next 指向 p，避免后续结点地址丢失。
   - 双向链表插入：在 p 后面插入 q，正确操作是 q->llink = p；q->rlink = p->rlink；p->rlink->llink = q；p->rlink = q。
   - 双向链表删除：删除结点 p（左右结点非空），需保证前驱结点的 rlink 指向后继结点，后继结点的 llink 指向前驱结点，如 p->llink->rlink = p->rlink；p->rlink->llink = p->llink。