《计算思维》考试要点

一 计算思维概述

1. 大致了解计算思维的概念；

**思维**

**思维**是认识客观世界和思考问题的方式，是生产思想的方法，它用于反映，解释，抽象客观世界，并能够预测未来的世界

**计算思维**

**计算思维**是运用计算机科学的基础概念去求解问题、设计系统和理解人类的行为，它选择合适的方式去陈述一个问题，对一个问题的相关方面建模并用最有效的办法实现问题求解。

1. 大致了解计算机的概念；

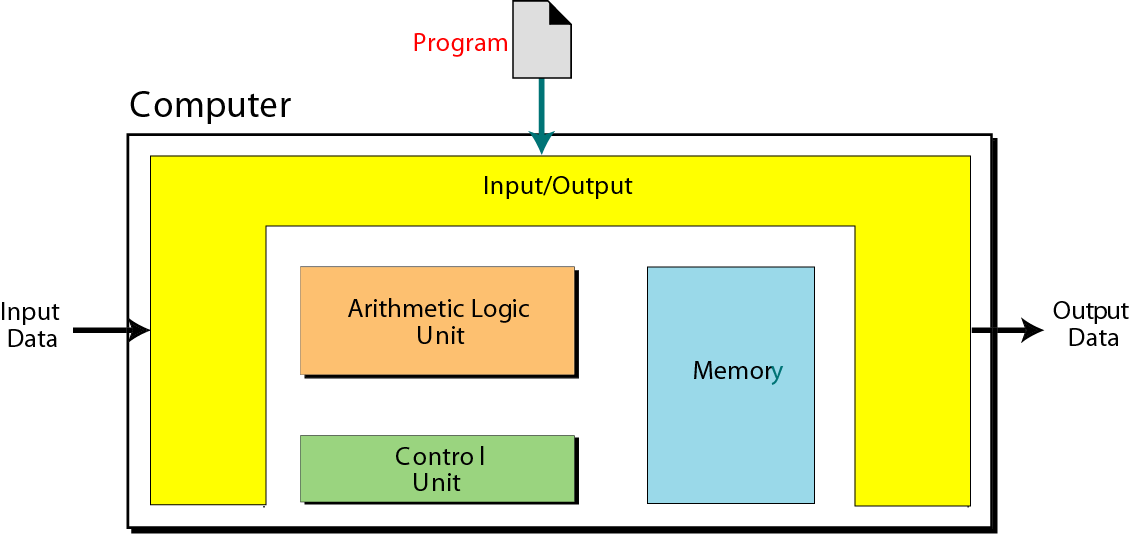
**计算机**可以看做是一个**数据处理模型**，它根据事先存储在存储器中的一组指令的要求，接收输入的数据、处理输入的数据、 并且输出数据处理结果。



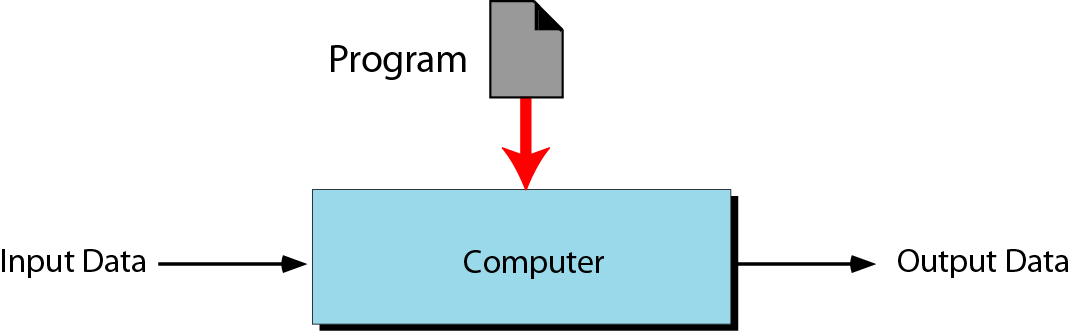
**Data processor model**

1. 详细了解冯诺依曼模型；

数学家**冯.诺依曼(von Neumann)**提出了计算机制造的**冯.诺依曼模型**，即采用二进制逻辑、程序存储执行以及计算机由**运算器**、**控制器**、**存储器**这三个核心组件，再加上**输入/输出设备**组成。



**Programmable data processor model**

从电子计算机诞生开始到现在，除了早期少数的计算机外，绝大多数包括当前正在使用的计算机都构建在冯诺依曼模型之上，所以现代计算机也可以看做是**可编程的数据处理模型**

1. 世界上最早使用割圆术求圆周率的数学家是谁？大致了解其求圆周率的思路；

目前所知最早用割圆术求圆周率是**阿基米德**，他分别计算圆的外切正六边形及内接正六边形的边长，以此计算圆周率的上限及下限，之后再将六边形变成十二边形，继续计算边长……，一直计算到正**96**边形为止。他计算出的圆周率的值在**223/71**和**22/7**之间。

此后的托勒密等人也使用割圆术来计算圆周率，但他们是单向迫近，结果不如阿基米德的双向迫近精确。

5.中国最早使用割圆术求圆周率的数学家是谁？大致了解其求圆周率的思路。

中国最早使用割圆术求圆周率的是魏晋时期的数学家**刘徽**，他也是从正六边形开始计算的，不同于阿基米德的是，刘徽是以面积来计算圆周率，他通过一种快速计算方法，把圆内接正多边形的面积一直算到了正3072边形，并由此求得了圆周率的值在**3.1415**和**3.1416**之间。

南北朝时期的数学家**祖冲之**在刘辉割圆术的基础上做了进一步研究，将圆周率的值精确到了**3.1415926**与**3.1415927**之间

1. 计算机软件分为系统软件与应用软件，大致区分哪些是系统软件，哪些是应用软件。

**1.2.3 计算机软件（Computer software）**

是指计算机系统中的程序及其文档，程序可以看作各类指令的序列；文档是为了便于了解程序所需的阐明性资料。

计算机软件一般可分为**系统软件**与**应用软件**，系统软件的典型代表是操作系统（Windows、Unix、Linux等）；应用软件则丰富多样，如QQ、Word、Phontshop等都属于应用软件。

7. 了解计算机发展的三个历史阶段：机械计算机、早期电子计算机、现代计算机；了解现代计算机历经的几代发展：第一代是电子管（真空管）时代，第二代是晶体管时代，第三代是集成电路时代，第四代开始（包括第五代）是超大规模集成电路时代。

**1.4.1 机械计算机器（1930年以前）**

**1.4.2 早期电子计算机的诞生（1930-1950）**

主要特点是采用电子管作为基本电子元器件，体积大、耗电量大、寿命短、可靠性低、成本高。没有系统软件，用机器语言和汇编语言编程。计算机只能在少数尖端领域中得到运用，一般用于科学、军事和财务等方面的计算。

典型代表：

**ABC（Atanasoff Berry Computer） Z1**  **Mark I**  **Colossus**

**ENIAC（Electronic Numerical Integrator and Calculator）**

* **Colossus（巨人计算机 ）**

英国数学家阿兰.图灵在1936年发表的论文中，提出了一种计算机抽象模型，利用这种计算机可用一些简单的机械动作实现推理。这种计算机也称**图灵机。**

1943年，图灵设计了**Colossus（巨人计算机）**，主要用于第二次世界大战破译德国的密码。

* **ENIAC（Electronic Numerical Integrator And Calculator）**

第一台电子计算机ENIAC诞生于1946年，由莫奇利与埃克特设计。编程通过插孔和开关实现，计算速度为5000 次/秒，  
输入用卡片，输出用指示灯，占据空间约1000平方英尺。

**1.4.3 现代计算机的诞生（1950-现在）**

第一代计算机（大约1950-1959年）

主要特征：

* 采用真空管作为主要元器件
* 只有专家才会使用
* 主要用于商用，体积大价格贵，只有大机构才能负担得起。

典型代表：

* **EDVAC（Electronic Discrete variable Automatic Computer）**

第一台使用二进制系统的计算机，诞生于 1949年。

* **EDSAC（Electronic Delay Storage Automatic Calculator）**

第一台基于冯.诺伊曼模型(存储程序)的计算机，诞生于 1950年。

* **UNIVAC (Universal Automatic Computer)**

第一台商业计算机，诞生于 1951年。

第二代计算机（大约1959-1965年）

主要特征：

* 采用晶体管作为主要元器件
* 与第一代相比体积变小
* 高级程序设计语言的出现使得编程变得容易
* 成本下降中小企业也能负担得起

典型代表：

* **PDP-1**
* **CDC6600**

第三代计算机（大约1965-1975年）

主要特征：

* 采用集成电路作为主要元器件
* 小型计算机面向市场
* 软件业开始产生
* 成本进一步下降

典型代表：

* **Cray-1**
* **PDP-8** &**11**
* **IBM360**

第四代计算机（大约1975-1985年）

主要特征：

* 采用超大规模集成电路作为主要元器件
* 微型计算机问世并步入家庭
* 开始研究计算机网络
* 成本进一步下降

典型代表：

* **Apple I**
* **Apple II**

第五代计算机（大约1985-现在）

主要特征：

* 个人电脑以及掌上电脑出现
* 多媒体技术以及虚拟现实得到广泛应用
* 输入输出设备更加多样化
* ……

二 定位数系 与 数据存储

1.进位制转换

（1）二进制数与十进制数（整数和小数）之间的互相转换，参考题目见例2-1到例2-4；

（2）二进制数与十进制数（整数和小数）之间的互相转换，参考题目见例2-5到例2-8；

（3）十六进制数与十进制数（整数和小数）之间的互相转换，参考题目见例2-9到例2-12；

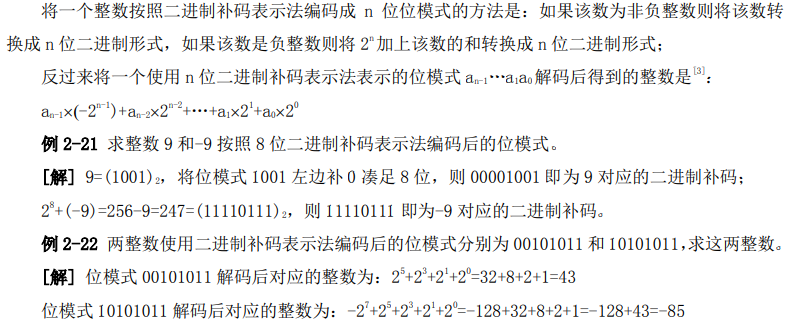
（4）二进制数与八进制数（整数和小数）之间的互相转换，参考题目见例2-13与例2-14；

（5）二进制数与十六进制数（整数和小数）之间的互相转换，参考题目见例2-15与例2-16。

参照笔记 对应练习

2.整数存储

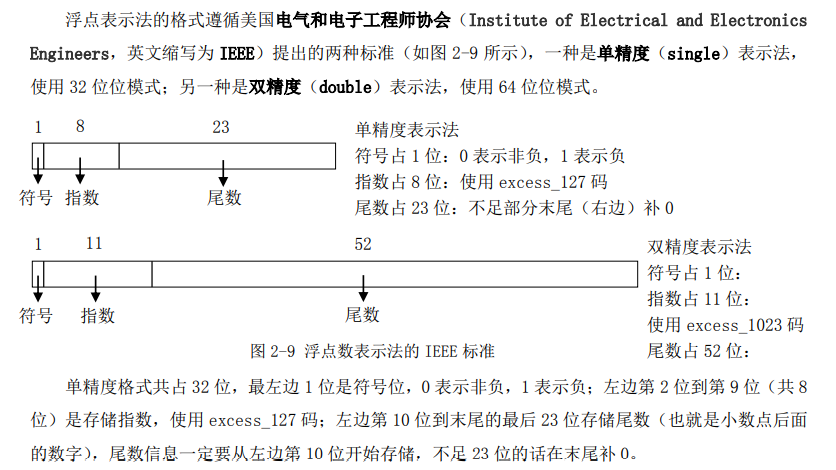
补码表示法的编码与解码，参考题目见例2-21与例2-22；

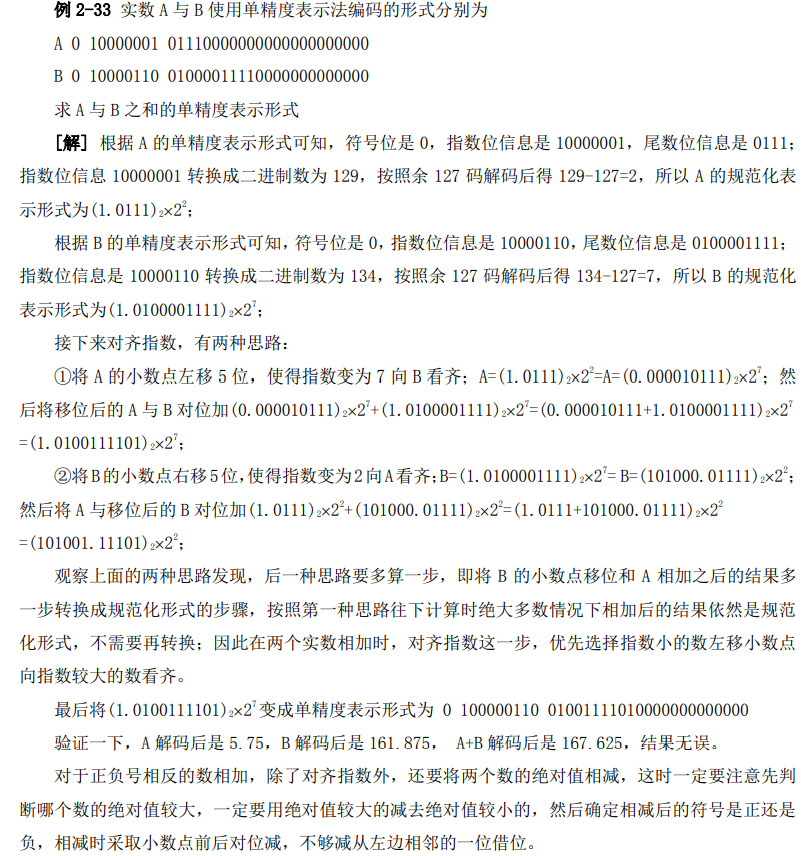


比特位之间的加法运算法则为 0+0=0，0+1=1，1+0=1，1+1=10，

3.浮点数存储

单精度表示法的编码与解码，参考题目见例2-30到例2-32；



4.其它数据的存储

（1）了解ASCII的编码与解码，了解汉字编码（不涉及到计算题）

（1）ASCII 美国国家标准学会（American National Standards Institute，英文缩写为 ANSI）在 1967 年发 布了美国信息交换标准代码（American Standard Code for Information Interchange，英文缩写为 ASCII），它包含了 128 个字符在计算机中存储时对应的位模式编码，编码方式很简单，给每个字符分 配一个十进制自然数编号，从 0 开始到 127 为止，然后将这些自然数编号转换成二进制形式，即为每 个字符对应的 ASCII 码。128=27，所以理论上来说存储这 128 个字符只需要 7 位即可，但是在计算机系 统中约定了 8 个比特位（也就是 1 个字节）是存储数据的最小单位，所以 ASCII 中的每个字符也采用 8 位存储格式，每个字符占据 1 个字节的存储容量。ASCII 中的这 128 个字符里面包含全体英文字母的大 小写，0-9 的数字，还有一些基本的标点符号以及数学运算符号，计算机键盘上大部分按键对应的字符 都在 ASCII 码表里可以找到，完整的 ASCII 码信息如表 2-13 所示。 表 2-13 中每个字符对应的编码都是转换成十进制后的形式，其中前 32 个（0-31）和最后 1 个（127） 字符是控制字符或通信专用字符，控制字符包括 LF（换行）、CR（回车）、FF（换页）、DEL（删除）、BS （退格)、BEL（振铃）等，通信专用字符包括 SOH（文头）、EOT（文尾）、ACK（确认）等；这些字符通 常都是不可显示的字符，也就是说输出时屏幕上看不到这些字符，但是这些字符会控制光标的位置， 比如 LF（换行）符，会让光标从上一行移到下一行。ASCII 码表中其它字符都是可以显示的字符，包 括 52 个英文字母（大小写各 26 个）、0-9 的数字、基本数学运算符和一些标点符号等。 有了 ASCII 码表英文文本的存储就比较简单了，直接将文本中的每个字符转换成对应的 ASCII 码 即可。按照这种格式保存在计算机中的文件被称为文本文件（text file），最常见的文本文件是后缀 名为“.txt”的文件。8 位位模式可以对 256（2 8 =256）个字符进行编码，ASCII 里面只有 128 个字符，对应的 8 二进制 编码都是 0 开头的位模式，1 开头的位模式并没有使用到。所以 1981 年 IBM 公司发布了扩展的 ASCII 码表，增加了 128 个新的字符，包括了某些带重音的字元、一个常用于表示数学符号的小写希腊字母 表、一些块型和线状图形字元等。不过扩展的 ASCII 码表并非国际标准。

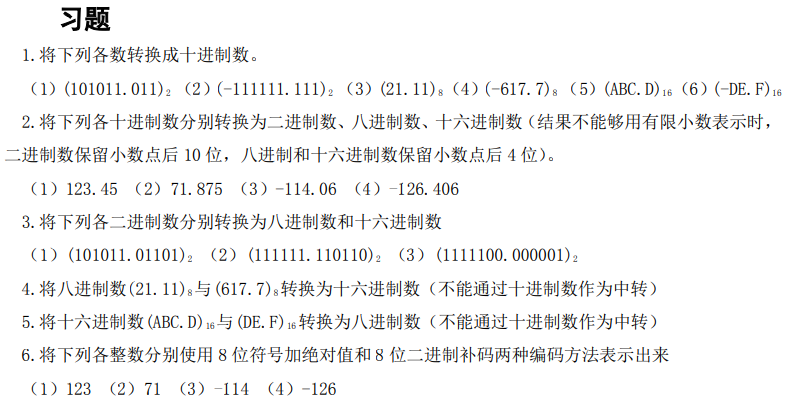
（2）汉字编码 ASCII 码解决了英文文本的编码问题，非英文的语言文字的编码问题则需要另想办法，和英文类似 由字母构成的语言在文本编码方面处理起来有一定优势，而汉字构成的中文则麻烦许多。常用的汉字 有几千个，给每一个汉字设计一个位模式编码是一项很繁琐的工作，我国国家标准化管理委员会 （Standardization Administration of China，英文缩写为 SAC）在 1981 年 5 月 1 日发布了《信息 交换用汉字编码字符集（基本集）》（简称 GB2312），现在常说的国标码就是指的这个编码字符集。不过 国标码并不是汉字在计算机中保存的编码形式，而且汉字的计算机化也不止存储需要编码，输入输出 也需要编码，下面简要介绍这些编码的来龙去脉。

（2）大致了解矢量图与位图的简单区分、多媒体数据、位(bit)、位模式、字节的概念。

计算机能够存储和处理的图片主要有 2 类：矢量图（vector graphic）和位图（bitmap）。矢量图 形使用直线和曲线来描述图形，这些图形的元素是一些点、线、矩形、多边形、圆和弧线等等，它们 都是通过数学公式计算获得的[5]。矢量图形在计算机处理的图形中所占比例较小，尤其是色彩多样的图 片不适合存储成矢量图的格式，接下来主要介绍位图的存储原理。

数据（data）可以简单地理解为对人类有用的信息，比如科学计算或实验中的各种结果以及新闻 报道中发布的文字、图片以及视音频这些都是数据的实例，计算机处理的数据从形式上可归纳为数字、 文本、图像、音频、视频这五种类型，计算机科学领域将这五类数据统称为多媒体数据（multimedia data）。

计算机存储的信息中，单个数字 0 或 1 被称为位（binary digit，英文缩写位 bit），因为 bit 直 译可读作比特，所以 binary digit 有时也被称为比特位，比特位是计算机系统中最小的存储单位，多 个比特位组成的数字串被称为位模式（bit pattern），8 个比特位构成的位模式被称为字节（byte）。 因为计算机存储的容量是有限的，并且为了方便数据在计算机系统中的存取和处理，保存同类数据所 占用的比特位数也是相同的，每种类型的数据存储时占用多少比特位，不同的系统有不同的约定。在 计算机系统中约定字节（8 个比特位）是最小的存储单位，也就是说任何类型的数据保存在计算机中都 至少占据 1 个字节的存储容量（也称为存储空间），且都是字节的整数倍。

5.本章课后练习掌握第1题到第6题，最好每题都做一遍。

三.程序设计概述

1.计算机程序设计语言的发展历程大致上可分为哪几个阶段？大致了解每个阶段的程序设计语言的特性；

计算机程序设计语言的发展历程大致上可分为三个阶段：

机器语言 汇编语言 高级语言

**机器语言**

**机器语言**是用二进制代码表示的计算机能直接识别和执行的一种**机器指令的集合**。它是计算机的设计者通过计算机的硬件结构赋予计算机的操作功能，**机器语言编写的程序等于直接向CPU发指令， CPU可以直接处理和执行机器指令**。

早期的计算机原始而简陋，没有今天的键盘、鼠标和显示器等设备，所有的指令和数据都以打孔带的形式输入输出

**汇编语言**

为了减轻机器语言带给程序员的不适应，人们对机器语言进行了升级和改进：用一些容易理解和记忆的单词来代替特定的指令。通过这种方法，人们很容易去阅读程序或者理解程序正在执行的功能，对现有程序的修改和维护都变得更加简单方便，这种语言就是我们所说的**汇编语言**。

由于汇编语言使用了人类熟悉的助记符，计算机无法像机器语言编写的程序一样直接识别和执行，必须把汇编语言转换成能够被计算机识别和处理的机器语言。

**高级语言**

使用机器语言和汇编语言进行程序设计时，要求使用者必须了解计算机的硬件结构，而不同型号的计算机硬件结构是不一样的，这极不利于程序在不同型号计算机上的推广。

随着计算机技术的发展，计算机科学家开发出接近自然语言且与具体硬件结构无关的程序设计语言，使用这些语言编写程序可以不需要了解计算机的硬件细节，这样的程序设计语言被称为**高级程序设计语言**，简称为**高级语言**。

2.高级程序设计语言的典型代表有哪些？其中哪些是结构化程序设计语言？哪些是面向对象的程序设计语言？

高级程序设计典型语言代表有：Fortran、Pascal、C、C++、Java、Python等。

高级语言也分为**结构化程序设计语言**与**面向对象的程序设计语言**， Fortran、Pascal、C 是**结构化程序设计语言**， C++、Java、Python 是**面向对象的程序设计语言**。

3.详细了解编译过程、解释过程、源程序（源代码）、目标程序（目标代码）、可执行文件等概念；

**编译与解释**

高级语言编写的程序也被称为**源程序**或**源代码**，需要经过转换成机器语言编写的程序后才能被计算机执行，转换过程有**编译**与**解释**两种。两者的区别在于，**编译**是一次性将高级语言编写的程序全部转换成机器语言编写的程序后再执行，**解释**则是将高级语言编写的程序逐条转换成机器语言编写的程序，每转换一条后就执行一条。

实现编译功能的程序被称为**编译器**，实现解释功能的程序被称为**解释器**。

常用的高级语言中，C和C++是编译型语言，Python是解释性语言。Java即是编译性也是解释性语言。

源程序(或源代码)转换后的机器可执行的程序也被称为**目标程序**或**目标代码**，计算机可执行的程序保存在后缀名为“exe”的文件中，所以后缀名为“exe”的文件也被称为**可执行文件**。

4.Python语言需要掌握的要点：

（1）输入语句input与输出语句print的使用；

（2）python包含的常用数据类型有字符串(str)、列表(list)、整数(int)、浮点数(float)、复数(complex)、布尔型(bool)、元组(tuple)；

（3）赋值运算符(=)、算术运算符(+,-,\*,/,%,\*\*,//)、比较运算符(==,!=,>,<,>=,<=)、逻辑运算符(and,or,not)、位运算符(&,|,~,^,<<,>>)的使用；位运算符是把数字看作二进制来进行计算的，位运算可以理解为针对比特位的运算，常用位运算符如下表所示：

|  |  |
| --- | --- |
| **位运算符** | **说明** |
| & | 按位与（两个比特位值均为1时结果为1，否则结果为0）  1&1=1，1&0=0，0&1=0，0&0=0 |
| **|** | 按位或（两个比特位值均为0时结果为0，否则结果为1）  1**|**1=1，1**|**0=1，0**|**1=1，0|0=0 |
| **^** | 按位异或（两个比特位值不同时结果为1，相同时结果为0）  1**^**1=0，1**^**0=1，0**^**1=1，0**^**0=0 |
| ~ | 按位取反，~1=0，~0=1 |
| **<<** | 左移，x << y表示x的二进制向左移动y位，低位补0 |
| **>>** | 右移，x >> y表示x的二进制向右移动y位，高位补0 |

假定a=22，其二进制为(0001 0110)2；b=35，其二进制为(0010 0011)2；那么：

a & b =(0001 0110)2 & (0010 0011)2= (0000 0010)2 = 2

a **|** b = (0001 0110)2**|** (0010 0011)2= (0011 0111)2 = 55

a **^** b = (0001 0110)2**^** (0010 0011)2= (0011 0101)2 = 53

**~**a =**~** (0001 0110)2= (1110 1001)2=-21，a的二进制各位取反之后最高位变成1，所以是负整数，按照补码解码后得到结果是-23。

a<<2 =(0001 0110)2 <<2= (0101 1000)2 = 88

a>>2 =(0001 0110)2>>2= (0000 0101)2 = 5

（4）列表的创建、添加、删除与复制；

（5）if语句、if-else语句、if-elif-else语句的使用；

（6）for语句与列表或range()函数结合实现循环以及while循环的使用；

（7）使用break语句终止循环；

（8）循环结束后的else语句执行；

（9）顺序、选择、循环三种程序控制结构的组合设计。

四.算法简介

1.算法有哪几个特性？

**算法**可以简单地理解为求解某个问题的**能行可计算过程**。算法最早是数学中的概念，计算机出现后延伸到计算机科学领域，使用计算机编程求解某问题的算法具备如下五个特性：

（1）有限性(有穷性)：一个算法必定要在有限的步骤内终止；

（2）确定性：一个算法的每一个步骤必需有精准的定义不会出现歧义；

（3）输入：一个算法有零个或多个输入；

（4）输出：一个算法至少有一个输出；

（5）能行性：一个算法的每一个步骤都是可执行的。

2.机器学习算法有哪两大类？这两大类算法的区分标准是什么？

机器学习算法主要分为两大类：**监督学习**和**无监督学习**。

**监督学习**是指学习数据本身被进行了**人工标注**，算法利用数据训练模型，然后用模型预测那些没有标注的数据，给出预测结果。例如，利用已经人工标注好猫和狗的图片各1000张，训练一个模型，可以让计算机识别没有见过的新图片，判断它是猫还是狗。

**无监督学习**是指学习数据本身是**没有人工标注**的，算法根据数据本身的特点给出预测结果。例如有猫狗图片各1000张，但并没有标明哪些是猫哪些是狗，让计算机自动把这些图片分成两类。

3.使用监督学习算法解决的问题分哪两大类？这两大类问题的区分标准是什么？有哪些典型的监督学习算法或问题？

使用**监督学习**算法解决的问题又可以分为两类：**回归问题**与**分类问题**。

**回归问题**指的是预测结果为连续值的问题，如线性回归、股票价格预测、天气预报等。

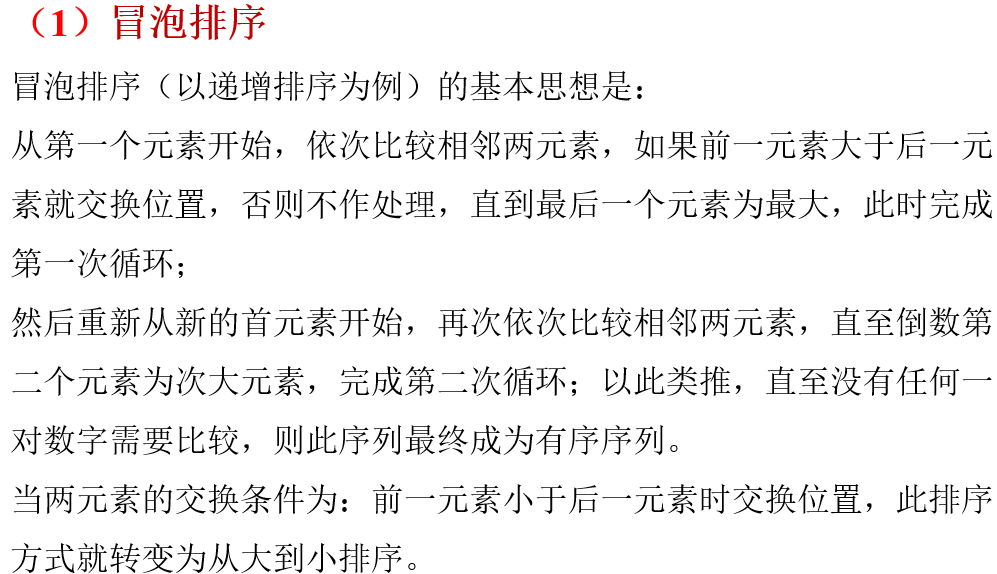
**分类问题**指的是预测结果为离散值的问题，如图片分类、人脸识别、神经网络分类等。目前流行的**神经网络**算法因其网络层次较多，也被称为**深度学习**算法。有一种解决分类问题的算法叫做**逻辑斯蒂回归算法**，虽然名字中有**回归**，它实际上是一种**分类算法**。

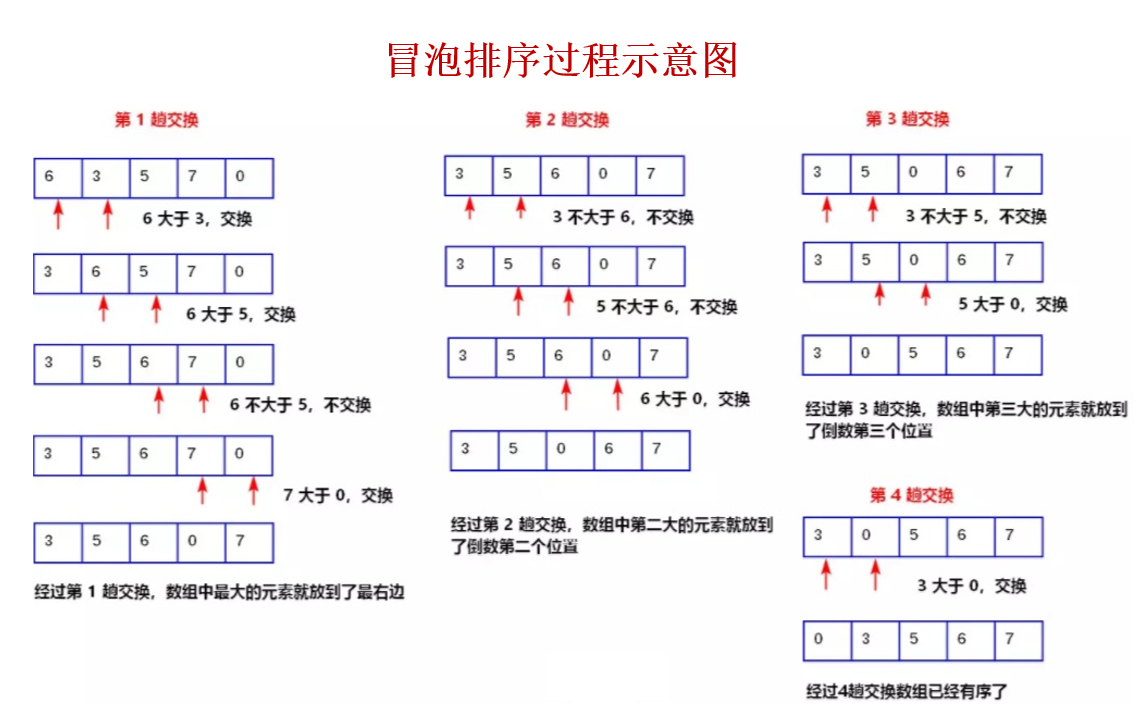
**监督学习算法**使用**人工标注好的数据**得到模型的过程称为**训练**，用模型再去对没有人工标注的数据进行分类或回归值的过程称为**预测**。监督学习算法处理的数据又通常被分成**训练集**和**测试集**，训练集用于构建模型，测试集用于测试模型的效率。

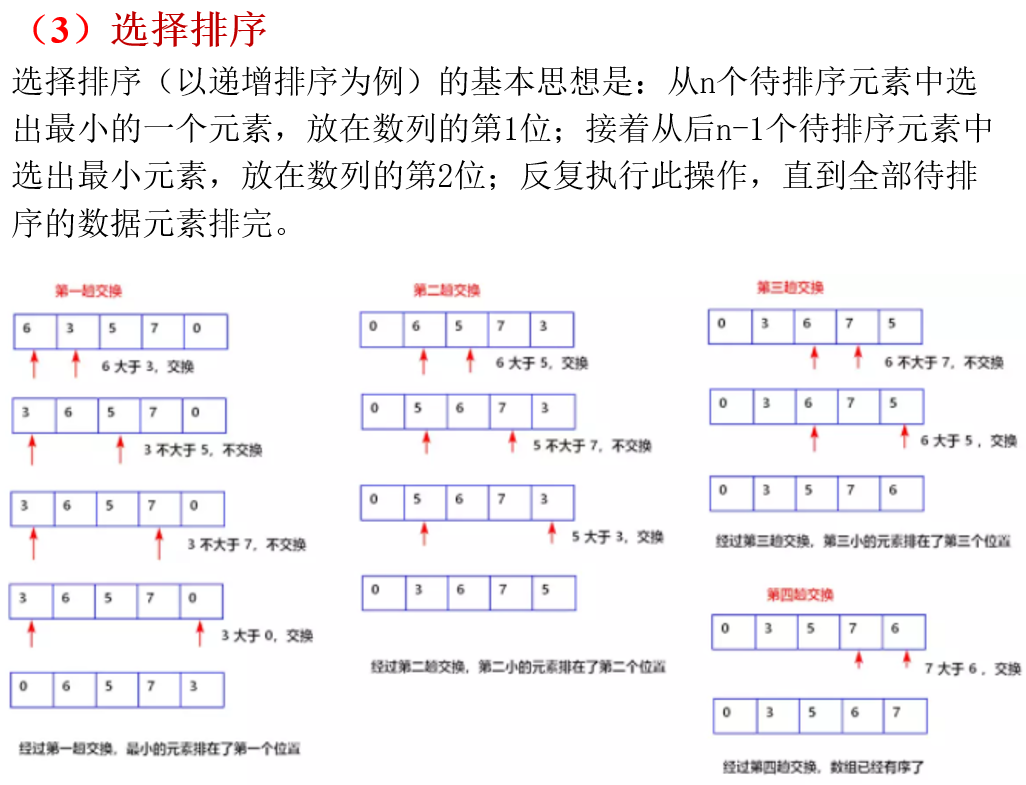
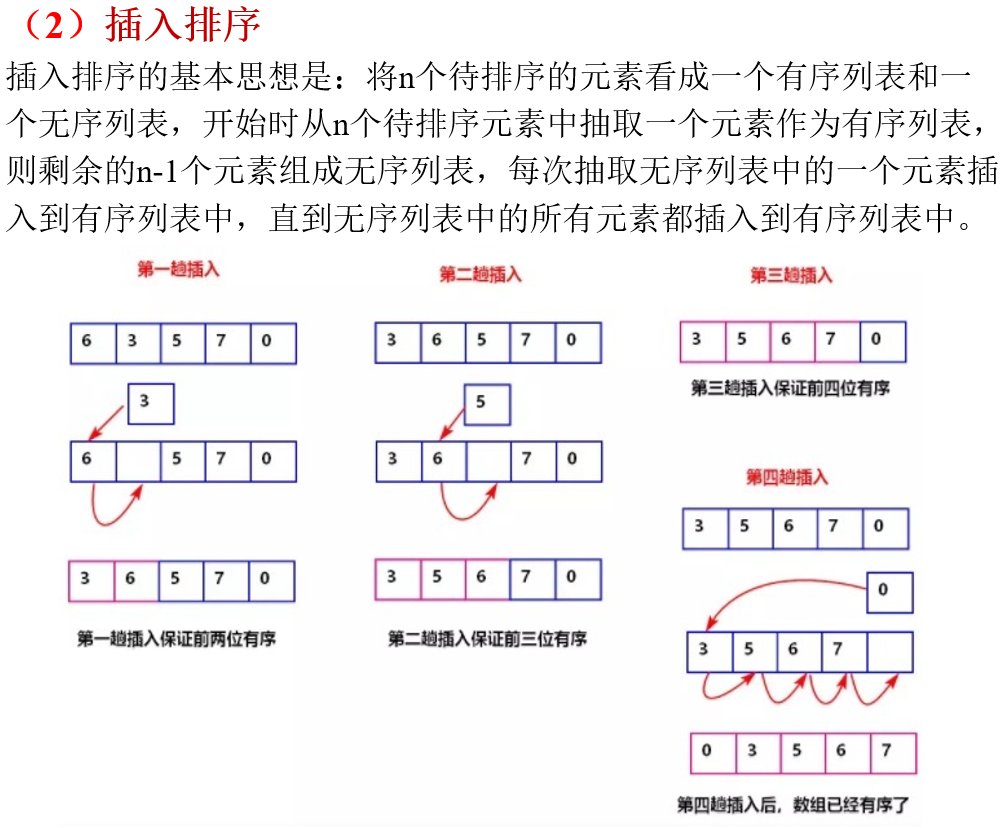
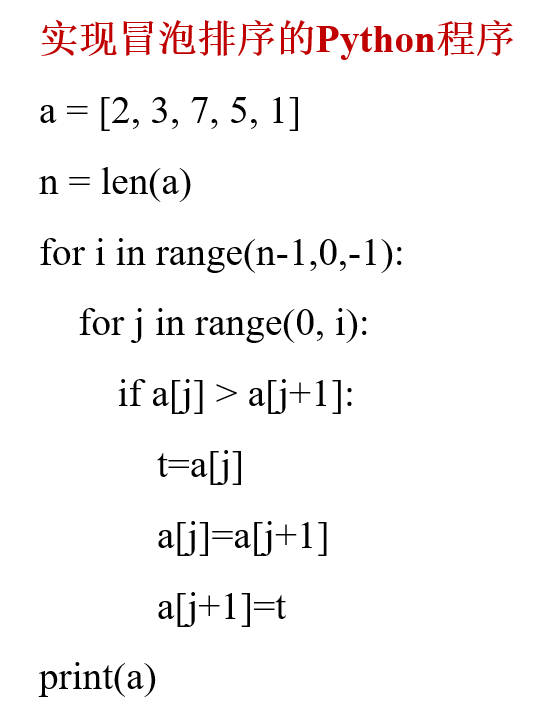
4.无监督学习算法的典型问题是什么？

使用**无监督学习**算法解决的问题中最典型的是**聚类问题**，它可以简单地描述为：根据在数据中发现的描述对象及其关系的信息，将数据对象分组。下图是聚类问题的一个实例，左上角子图中黑点表示没有被人工标记的对象，其余三个子图为执行不同聚类算法之后的分类结果，相同颜色与形状的点被划分为一类。

5.掌握冒泡排序、选择排序、插入排序、顺序查找、二分查找的算法思路，以及每种算法的python编程实现。

1. 





**4.4 查找算法**

查找算法的某元素在列表中的位置，

**（1）顺序查找**

**顺序查找**的基本思想是：

从列表的起始位置开始扫描，依次将扫描到的元素值与待查找的元素值相比较，若找到相等的则表示查找成功并输出该元素在列表中的位置（可以输出该元素在列表中的下标，如果列表中有多个元素值与待查找的元素值相等，则输出位置最靠前的元素的下标）；

若扫描结束仍没有找到与待查找的元素值相等的元素，则表示查找失败（通常输出一个负整数如-1表示查找失败）。

输入一个整数，输出其在列表[1, 2, 32, 8, 17, 19, 42, 13, 0]中的位置（下标），如果该数不在列表中，则输出-1。

实现顺序查找算法的python代码如下：

a=[1, 2, 32, 8, 17, 19, 42, 13, 0]

n=len(a)

x=int(input("输入一个整数x="))

posi=-1 **#变量posi表示x在列表a中的位置，默认其不在a中**

for i in range(n):

if x==a[i]:

posi=i

print(posi)

**（2）二分查找**

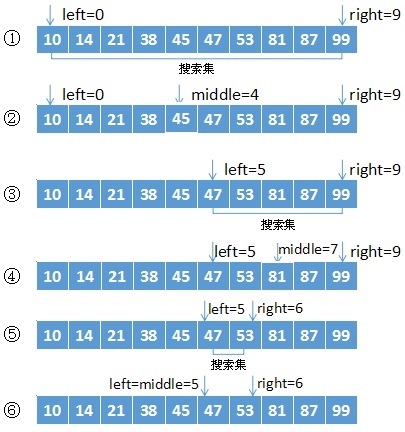
**二分查找**也叫做**折半查找**，也是查找某个元素在列表中的位置，不过二分查找针对的列表是有序表，列表元素值按照递增或递减顺序排列。**二分查找**的基本思想是：

查找过程从列表的中间元素开始，用待查找元素x先与列表最中间的元素m比较，m将列表分成两个子表，若x与m相等则查找成功；

若x小于m，则在前一个字表中继续查找x；

若x不小于m，则在后一个字表中继续查找x；进一步查找过程重复刚开始的步骤，递归进行下去，直到找到x在列表中的位置，或者发现x并不在列表中。

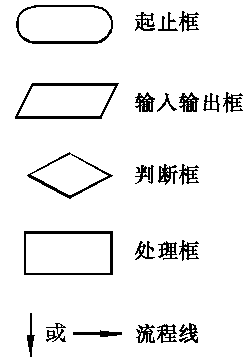
使用二分算法在列表中查找47的过程，查找成功，最终middle的值即为47在列表中的位置！



使用二分算法在列表中查找46的过程，最终right<left，查找失败！



6.掌握顺序、选择、循环三种程序控制结构的流程图表示，掌握将程序转变为流程图表示的方法。



**4.6 流程图**

结构化程序设计中的三种基本程序结构为**顺序结构**、**选择结构**、**循环结构**。任何复杂的程序都是由这三种基本结构组合而成。

**流程图**是表示程序结构的一种图形工具，其基本组成部分如右图所示。

**起止框**：每个流程图都有一个**开始框**和**结束框**；

**输入输出框**：对应输入输出语句；

**处理框**：对应赋值语句；

**判断框**：对应选择或循环时的判断条件；

**流程线**：对应语句执行的先后顺序。

**（1）顺序结构流程图**

只包含顺序结构的程序比较简单，通常只有输入、输出以及赋值语句。绘制流程图时每条语句对应的流程图符号，按照语句执行的顺序用流程线（箭头）连接起来，前后各加一个**开始框**与**结束框**即可。

**开始**

**t=a**

**输入a和b**

**结束**

**a=b**

**b=t**

**输出a和b**

下面顺序结构的程序对应的流程图如右所示。

a=int(input(‘输入a=’))

b=int(input(‘输入b=’))

t=a

a=b

b=t

print(a,b)

流程图中每个图形符号中的内容可以是一条语句，也可以是对多条语句的文字描述，只要描述清楚即可。

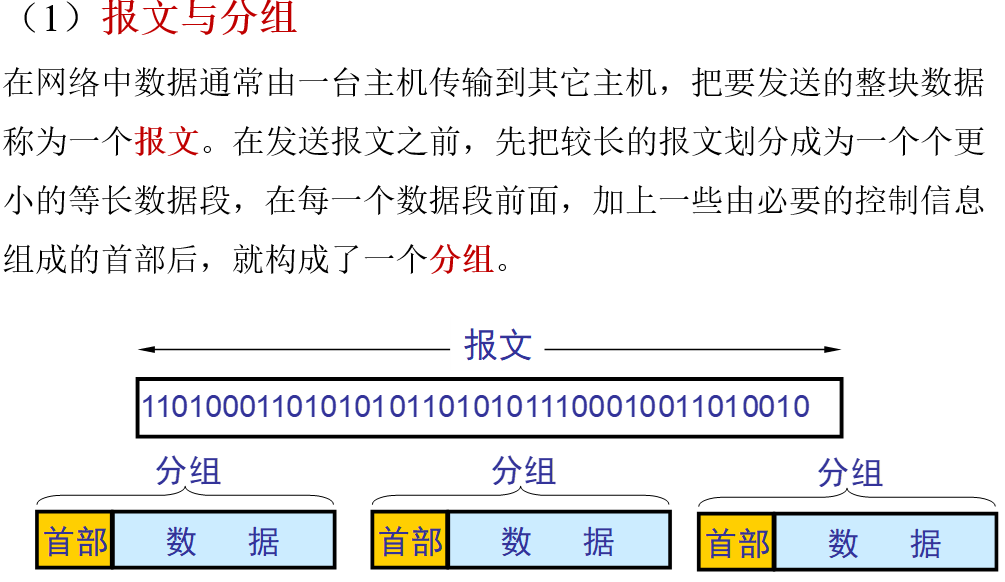
见算法简介

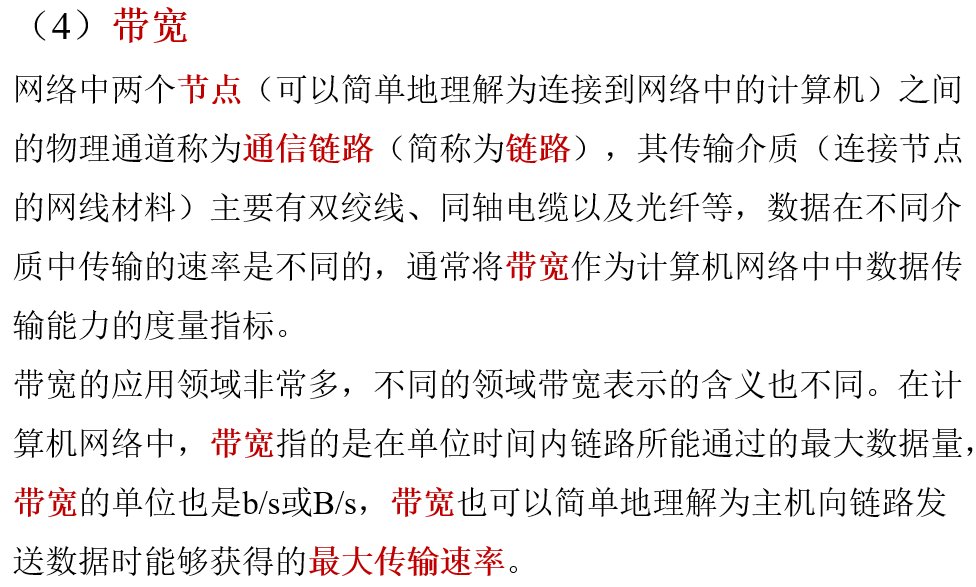
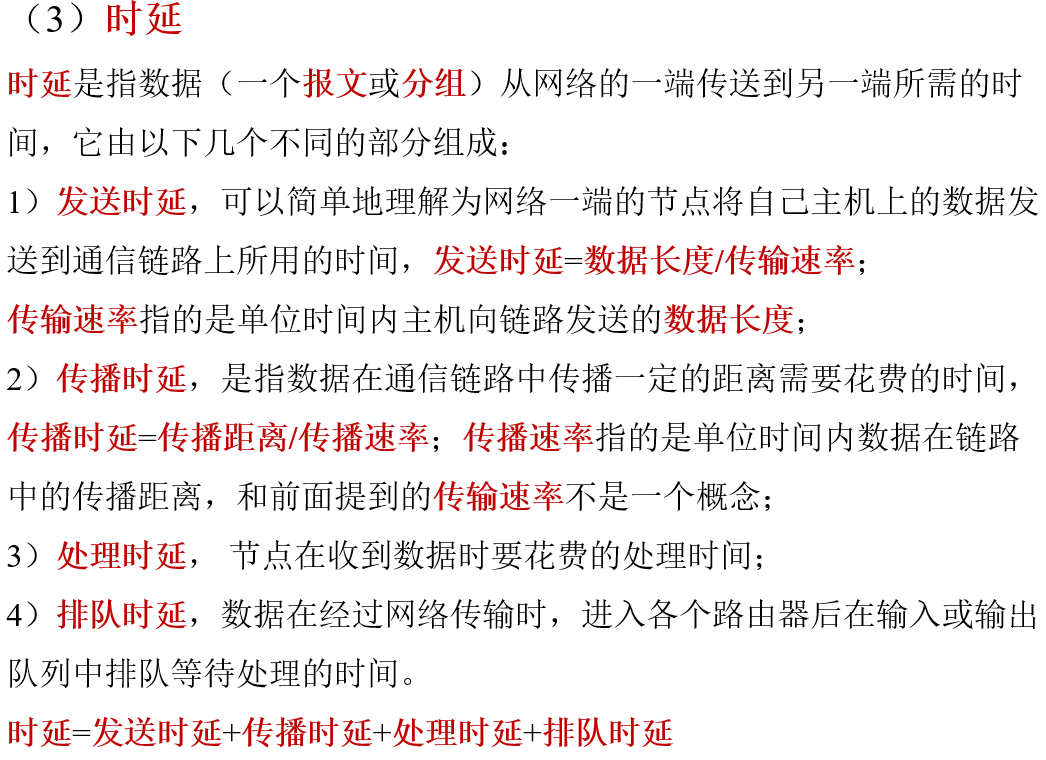
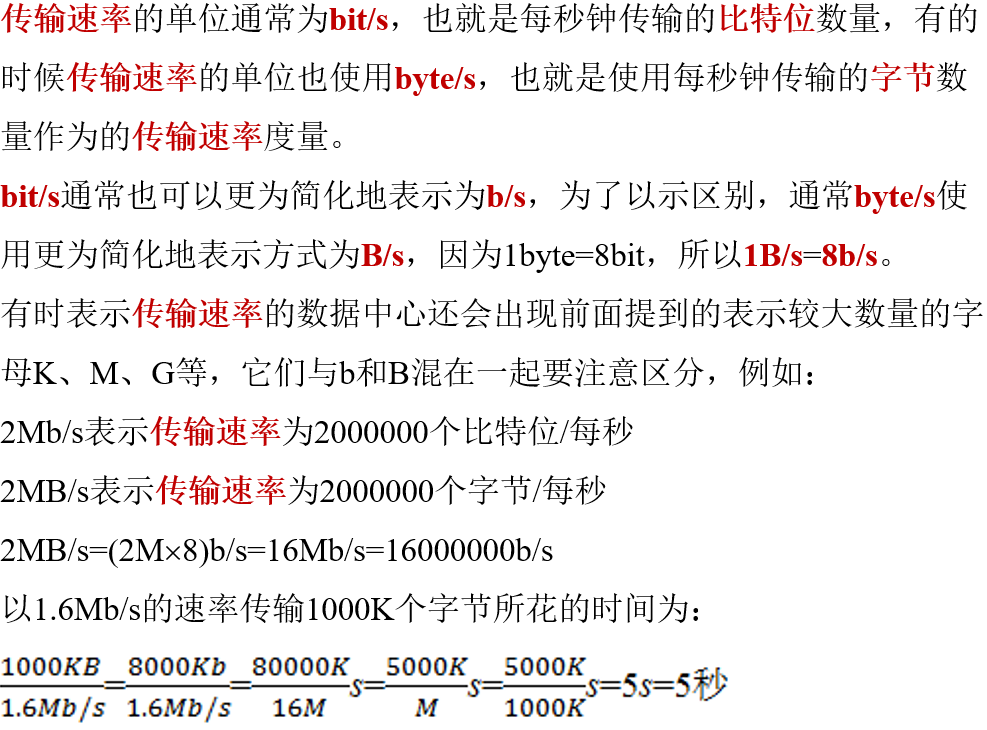
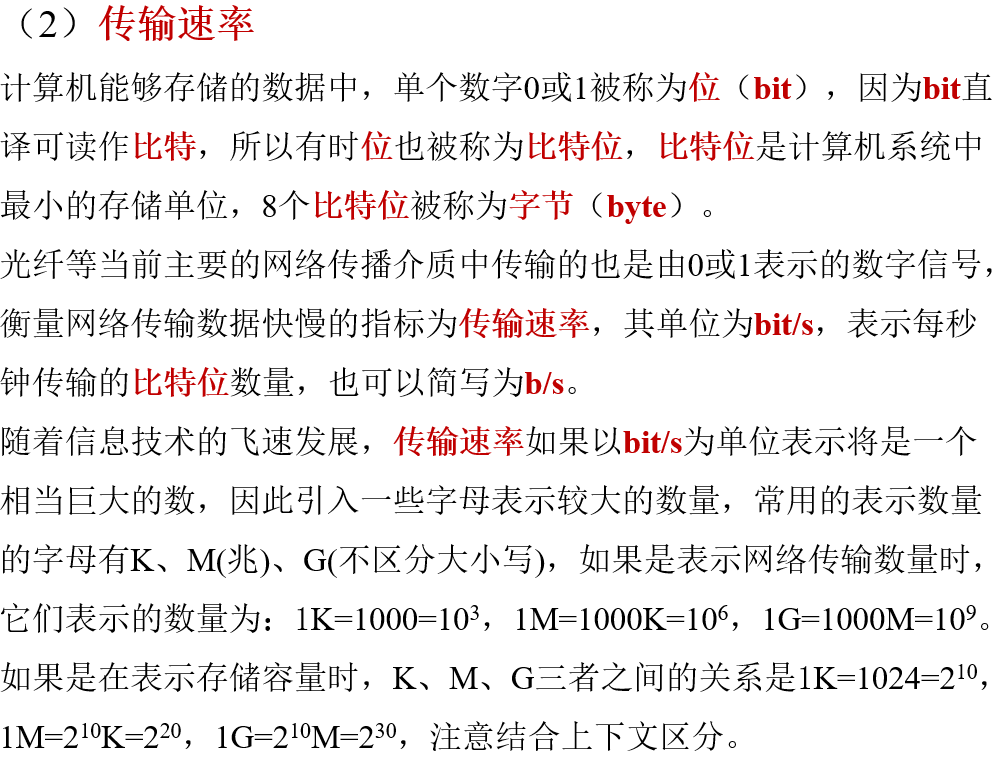
五.计算机网络简介

1.世界上公认的第一个计算机网络的名称是什么？

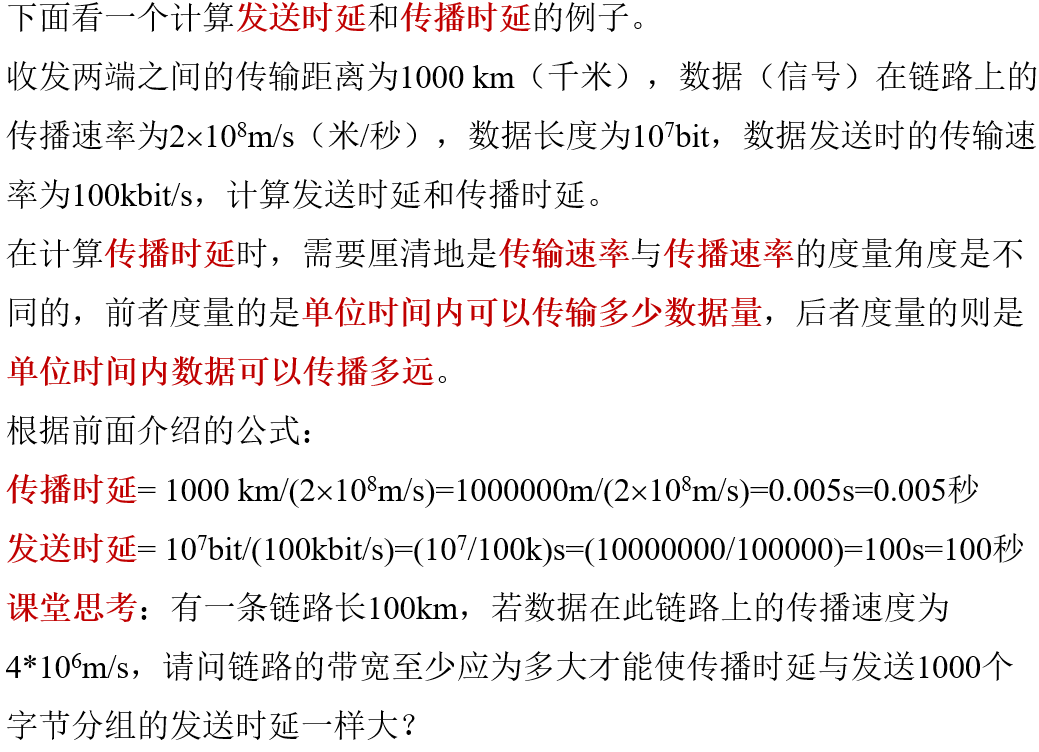
**ARPANET**

2.详细了解报文、分组、传输速率、带宽、时延（发送时延与传播时延）的概念；

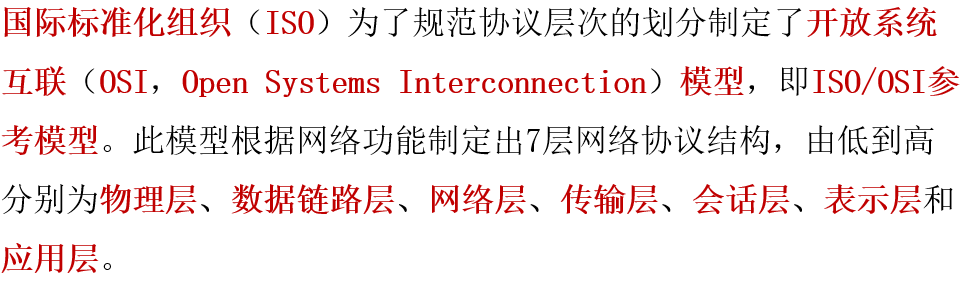


3.熟练掌握计算传输速率与发送时延的方法；

4.熟练掌握计算传播速率与传播时延的方法；



5. OSI模型的中英文全称分别是什么？它包含几层网络协议结构？每一层的名称是什么？



6.什么是网络协议？其三要素是什么？从网络研发开始并一直沿用至今的互联网基础通信协议是什么？它是由哪些协议组成的协议簇？

