信息学考复习指南

学考班级：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

行政班级：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

姓 名：\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

目录

[第一章 数据与信息 1](#_Toc91790344)

[**1.1感知数据** 1](#_Toc91790345)

[**1.2数据、信息、知识与智慧** 1](#_Toc91790346)

[**1.3数据采集与编码** 1](#_Toc91790347)

[**1.3.1数据采集** 1](#_Toc91790348)

[**1.3.2数制** 2](#_Toc91790349)

[**1.3.3字符编码** 2](#_Toc91790350)

[**1.3.4声音编码** 3](#_Toc91790351)

[**1.3.5图像编码** 3](#_Toc91790352)

[**1.3.6视频编码** 3](#_Toc91790353)

[**1.4数据管理与安全** 4](#_Toc91790354)

[**1.5数据与大数据** 4](#_Toc91790355)

[第二章 算法与问题解决 5](#_Toc91790356)

[**2.1算法的概念和描述** 5](#_Toc91790357)

[**2.2算法的控制结构** 6](#_Toc91790358)

[**2.3用算法解决问题的一般过程** 6](#_Toc91790359)

[第三章 算法的程序实现 6](#_Toc91790360)

[**3.1用计算机编程解决问题** 6](#_Toc91790361)

[**3.2Python语言程序设计** 6](#_Toc91790362)

[**3.2.1Python语法基础** 6](#_Toc91790363)

[**3.2.2Python控制结构** 10](#_Toc91790364)

[**3.2.3函数与模块** 11](#_Toc91790365)

[**3.3简单算法及程序实现** 12](#_Toc91790366)

[**3.3.1解析算法** 12](#_Toc91790367)

[**3.3.2枚举算法** 12](#_Toc91790368)

[第四章 数据处理及应用 13](#_Toc91790369)

[**4.1常见表格数据的处理** 13](#_Toc91790370)

[**4.2编程处理数据** 15](#_Toc91790371)

[第五章 人工智能及应用 15](#_Toc91790372)

# 第一章 数据与信息

**1.1感知数据**

1. 数据的定义：数据是对客观事物的符号表示，其表现形式包括图形符号、数字、字母、图像、视频、音频等。数字是最简单的一种数据。

数据不是现代社会所特有的，是自古就有的。如原始人在墙壁或石头上刻画图案就是最原始的数据。

1. 数据与生活：人们利用数据的同时，自身行为也产生着数据。
2. 数据与科学：数据的客观性为科学研究提供了可靠的依据。

**1.2数据、信息、知识与智慧**

1. 信息的定义：信息是用来消除随机不确定的东西。指数据、信号、消息中包含的意义。
2. 信息与载体区分：信息是看不见摸不着的。载体可以是实体的，也可以是非实体的。文字、图像、声音、二进制代码等都是信息的载体。U盘、磁盘、甚至我们的大脑也是信息的载体。文字、图像里面所表达的意义才是信息。
3. 信息的特征：载体依附性、时效性、共享性、可加工处理性、真伪性、价值性。

[载体依附性]

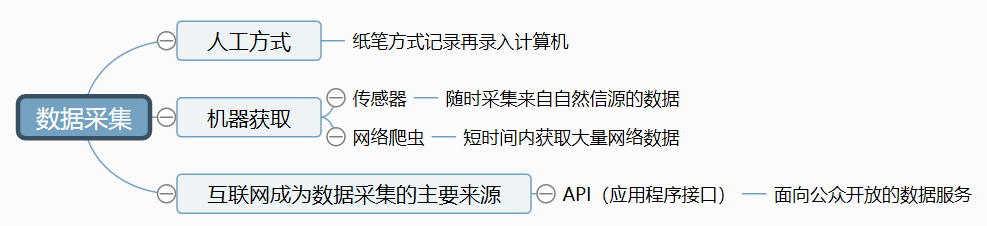
* 1. 信息不能独立存在，需要依附于一定的载体。载体被破坏，信息也消失了。
  2. 同一个信息可以依附于不同的载体。
  3. 信息可以脱离它所反映的事物被存储、保存和传播。（历史事件已经发生过了，但是其信息可以被记录在史书上保留并传播；再如案件）

[价值性]

1. 信息的价值包括显性价值和隐性价值。
2. 信息的价值是相对的，对于不同的人群价值高低不同。
3. 数据与信息的关系：单纯的数据没有意义，将该数据进行处理、加工后以适当的形式呈现出来，我们就能够理解它所表示的含义了，这就是信息。可以简单地理解数据是原始素材，而信息是具体内涵。
4. 数据、信息、知识与智慧的关系：
   1. 信息是数据经过加工解释后产生的意义。
   2. 人们通过归纳、演绎、比较等手段对信息进行挖掘，将万千信息中有价值的部分沉淀下来，与已存在的人类知识体系相结合，形成知识。
   3. 智慧是在知识的基础上，通过认识的累积，而形成的对事物的认识、远见，体现为一种卓越的判断力。（简单的知识积累并不等同于有了智慧）（这个层次很难达到）

**1.3数据采集与编码**

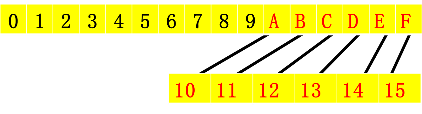
**1.3.1数据采集**



**1.3.2数制**

1. 进制：计数方法。一般称进位规则为逢n进一的数制为n进制。

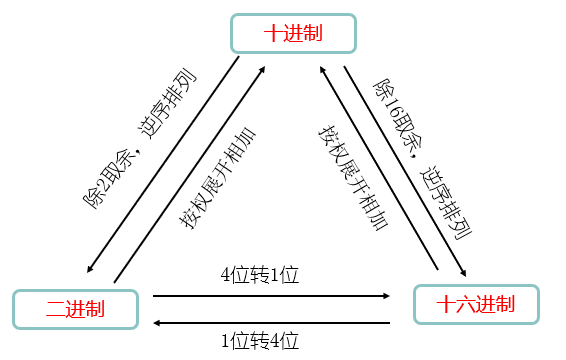
（其中十六进制后6个数码使用大写英文字母代替）



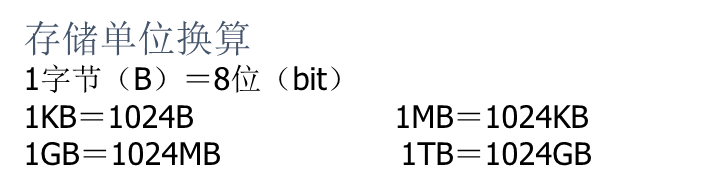
1. 进制的标识

|  |  |  |  |
| --- | --- | --- | --- |
| 进制 | 二进制 | 十进制 | 十六进制 |
| 字母标识 | 1011 B | 13 D | 3A H |
| 下标标识 | （1011）**2** | （13）**10** | （3A）**16** |

1. 二进制、十进制、十六进制的相互转换（相对较快的转换方法：8421法）

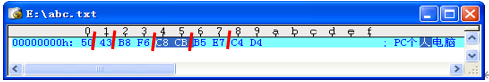
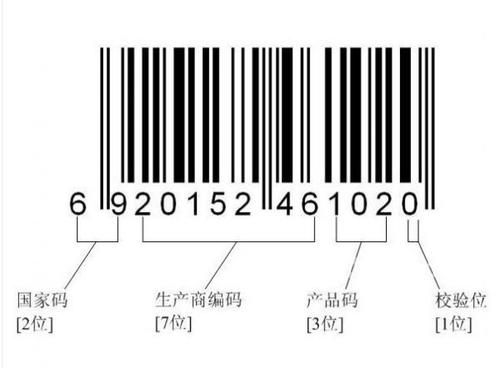


1. 计算机存储数据的最小单位：位bit（比特），存放一位二进制数据，即0或1。基本单位：字节Byte，8位二进制数据相当于一个字节。



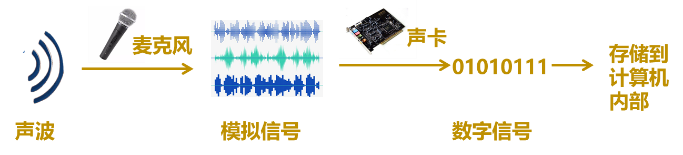
1. 信息编码：两种颜色，可以采用1位二进制编码：0和1。四种颜色可以采用2位二进制编码：00、01、10、11。总结：n位二进制可以表示**2n**种不同的编码。

**1.3.3字符编码**

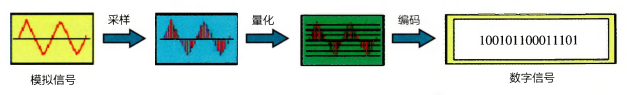
1. ASCII码：英文字符在计算机内部使用1个字节的ASCII码表示。基本的ASCII码除了大写英文字母和小写英文字母外，还有包括阿拉伯数字与一些特殊符号等共计128个字符，使用1个字节的低7位编码。0-127对应十六进制编码是00-7F。
2. 汉字编码：中文字符在计算机内部一般使用2个字节的汉字编码表示，如GB2312编码和GBK编码。
3. UltraEdit软件查看内码（此时查看到的是十六进制，但是存进计算机内部还是二进制）：
4. Unicode：将世界上所有的符号都纳入其中，解决乱码的问题。定长编码：两个字节表示一个字符。
5. [](https://image.baidu.com/search/detail?ct=503316480&z=undefined&tn=baiduimagedetail&ipn=d&word=%E6%9D%A1%E5%BD%A2%E7%A0%81&step_word=&ie=utf-8&in=&cl=2&lm=-1&st=undefined&hd=undefined&latest=undefined&copyright=undefined&cs=1569609805,3765640310&os=3266586726,3036786897&simid=19495693,731045860&pn=18&rn=1&di=198770&ln=1853&fr=&fmq=1631500377110_R&fm=&ic=undefined&s=undefined&se=&sme=&tab=0&width=undefined&height=undefined&face=undefined&is=0,0&istype=0&ist=&jit=&bdtype=0&spn=0&pi=0&gsm=0&objurl=https://gimg2.baidu.com/image_search/src=http://img003.file.rongbiz.cn/uploadfile/201406/04/13/25-32-68958.jpg&refer=http://img003.file.rongbiz.cn&app=2002&size=f9999,10000&q=a80&n=0&g=0n&fmt=jpeg?sec=1634092376&t=f4593a8aebc6dc4122dbd3a79a80d524&rpstart=0&rpnum=0&adpicid=0&nojc=undefined)UTF-8：Unicode的拓展，变长编码，1-4个字节表示一个字符。
6. 条形码和二维码

**1.3.4声音编码**

1. 模拟信号：以连续变化的物理量存在的信号。
2. 数字信号：在取值上是离散的、不连续的信号。
3. 模数转换过程（以声音数字化为例）：

 （录音：模数转换；放音：数模转换）

声卡即其中的模数转换器（ADC），关键转换过程有：采样、量化、编码。

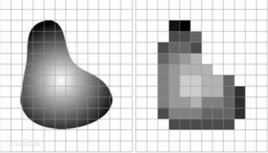
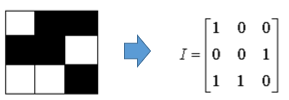


1. （WAV格式）声音存储容量计算公式：

采样频率（Hz）×量化位数（bit）×时间（s）×声道数÷8（单位：字节）

**1.3.5图像编码**

1. 数字图像分为矢量图形和位图图像两种。位图是通过“像素”来描述图像的，适合表现层次和色彩比较丰富的图像，所需存储空间较大，相机拍的都是位图，放大后会失真。矢量图是用一组指令集合或数学公式来描述的，放大旋转等操作不会失真。
2. 位图图像数字化过程：采样、量化、编码

[](https://baike.baidu.com/pic/%E5%9B%BE%E5%83%8F%E6%95%B0%E5%AD%97%E5%8C%96/4950387/1/9922720e0cf3d7ca69792e6ff91fbe096b63a91f?fr=lemma&ct=single) 

其中量化过程决定了使用多大范围的数值来表示图像采样之后每个像素的颜色信息，一般用二进制数表示，即颜色位深，256色即8位色。

1. （BMP格式）图像存储容量计算公式：

水平像素×垂直像素×颜色位深度÷8（单位：字节）

1. JPEG等格式的位图是经过一定程度压缩的图像，文件存储容量较小，不适用上述公式。

**1.3.6视频编码**

1. 静态的图像连续播放就形成视频。PAL制式的视频每秒钟播放25帧，NTSC制式的视频每秒钟播放30帧。帧数=帧频×时间。（一帧就是一副画面）
2. （AVI格式）视频存储容量：

水平像素×垂直像素×颜色位深度×帧频×时间÷8（单位：字节）

1. MPEG等格式的视频是经过一定程度压缩的视频，文件存储容量较小，不适用上述公式。

**1.4数据管理与安全**

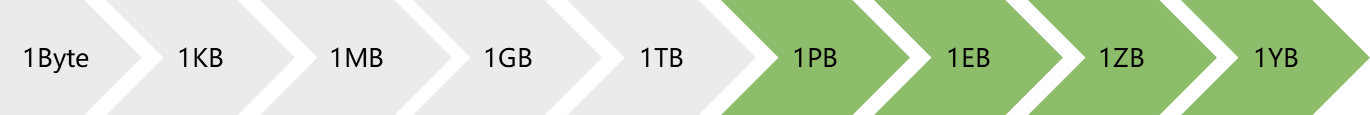
1. 数据管理：利用计算机硬件和软件对数据进行有效地收集、存储、处理和应用的过程。
2. 计算机数据管理的三个阶段：人工管理、文件管理、数据库管理。
3. 计算机管理数据的方式：树型目录、文件夹（Windows系统）
4. 文件拓展名：数据一般以文件的形式存储。根据编码规则的不同，文件的格式也不相同，用以区分不同类型的存储数据，如文本、图像、音频等。

|  |  |  |
| --- | --- | --- |
| **格式分类** | **扩展名** | **说明** |
| 文本文件 | txt | 纯文本文件 |
| doc/docx | word文件 |
| 图像文件 | bmp | 无压缩位图 |
| JPG | 常见位图 |
| gif/png/tif | 支持透明色 |
| 声音文件 | wav | 无压缩音频 |
| mp3 | 常用音频格式 |
| 视频文件 | avi/mpg/mp4/wmv | 常用视频格式 |
| 网页文件 | html/htm | |

1. 数据类型：结构化数据（二维表结构）、非结构化数据（各类格式的办公文档、文本、图片、网页、音频、视频等）、半结构化数据。
2. 威胁数据安全的因素：硬盘驱动器损坏、操作失误、黑客入侵、感染计算机病毒、遭受自然灾害等
3. 保护数据介质：异地容灾、数据备份（异地备份）、磁盘阵列
4. 提高数据本身安全：数据加密、数据校验（MD5/CRC/SHA-1等校验方法）

**1.5数据与大数据**

1. 大数据的定义：大数据代表着信息量大、速度快、种类繁多的信息资产，需要特定的技术和分析方法将其转化为价值。
2. 大数据的特征（4V理论）：
   1. Volume数据体量大：PB级别将是常态。



* 1. Velocity速度快：包括数据产生速度快和处理速度快
  2. Variety数据类型多：数据来源多，既有人工产生的，又有机器自动产生的。结构化数据、半结构化数据、非结构化数据共存是大数据的普遍现象。
  3. Value价值密度低：数据体量越大，价值密度越低（反比）。

1. 大数据思维：
   1. 采样 vs 全样：大数据要分析全体数据，而不是抽样数据。
   2. 精确 vs 模糊：大数据不追求数据的精确性，而是接受数据的混杂度。
   3. 因果 vs 关联：相关关系重于因果关系。
2. 大数据对社会的影响：
   1. 大数据已经渗透到社会各个层面，不断提高工作效率与生活便利性。
   2. 大数据让决策更精准。
   3. 大数据带来了新的就业机会。

④ 大数据带来信息泄露、数据安全、个人隐私甚至伦理道德方面的社会问题。

# 第二章 算法与问题解决

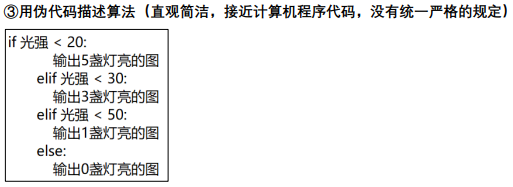
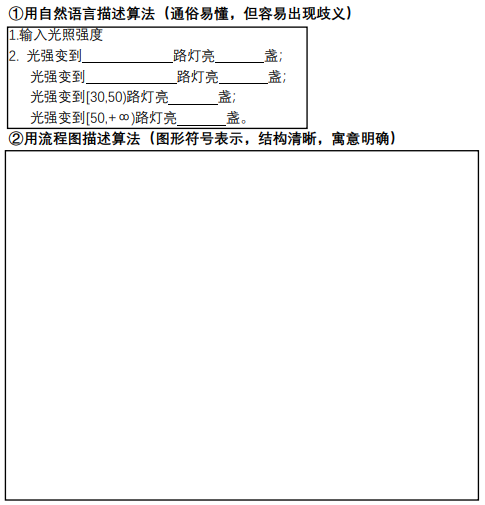
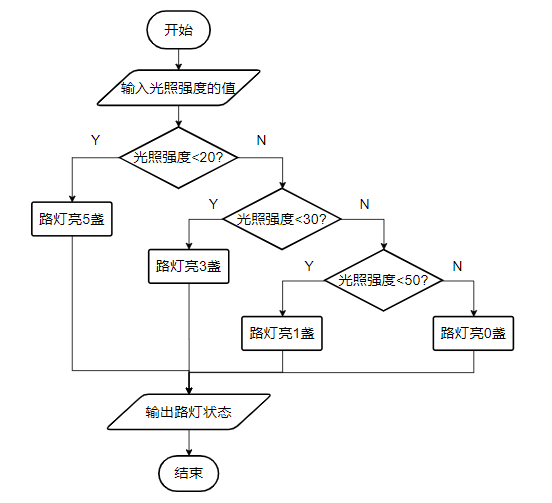
**2.1算法的概念和描述**

1. 算法的概念：

①广义地讲，算法指的是解决问题或完成任务的一系列步骤。

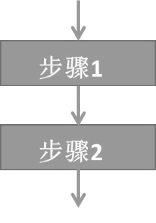
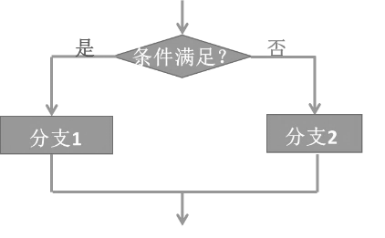
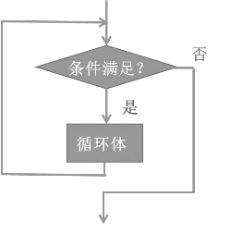
②在计算机科学领域内，算法指的是用计算机解决问题的步骤，是为了解决问题而需要让计算机有序执行的、无歧义的、有限步骤的合集。

1. 算法的特征：有穷性、可行性、确定性、0个或多个输入、1个或多个输出
2. 算法的要素：数据、运算、控制转移
3. 算法的描述方法：



**2.2算法的控制结构**

1. 算法的三种基本控制结构：顺序结构、分支结构（也叫选择结构）、循环结构

△区分：分支结构只判断了一次，循环结构往往会判断多次。

1. 计数器：常用于控制循环的次数，如i ← i+1。
2. 累加器：累加同样类型的数据，如初始状态sum=0，在循环结构中令sum ← sum+i。
3. 累乘器：累乘同样类型的数据，如初始状态sum=1，在循环结构中令sum ← sum×i。

**2.3用算法解决问题的一般过程**

抽象与建模 → 设计算法 → 描述算法

# 第三章 算法的程序实现

**3.1用计算机编程解决问题**

1. 用计算机编程解决问题的一般过程

抽象与建模 → 设计算法 → 编写程序 → 调试运行程序

1. 编程语言的发展：机器语言 → 汇编语言 → 高级语言

①机器语言：由“0”和“1”组成的二进制指令序列。

②汇编语言：用“英文缩略词”和“助记性符号”代替二进制指令。

③高级语言：使用人们熟悉的“数学语言”和“自然语言”来编写程序。

Python语言是高级语言。

**3.2Python语言程序设计**

**3.2.1Python语法基础**

1. 基本数据类型

|  |  |  |
| --- | --- | --- |
| **数据类型名** | **表现形式** | **关键字** |
| 整型 | 数学中的整数（十进制），如1，-89，0等，十六进制数会有0x前缀，例如0xff00 | int |
| 实型 | 数学中的实数（也叫浮点数），如3.14，-9.01等，用科学计数法表示的实数有，如0.000012可以写成1.2e-5，表示1.2×10**-5** | float |
| 字符串型 | 用单引号，双引号，三引号表示，如’x’，“this is sting”，’’’这是一个字符串！’’’ | str |
| 布尔型 | 只有两种值True和False，表示逻辑判断的结果 | bool |

△注意：5和5.0在数学意义上是一样的，但是在python中是不同类型的数据，一个是整型（int），一个是实型（float）。

1. 基本命令：

①输入数据input()

•获取用户输入的字符串。

• 圆括号内不填写内容，若填写，一般是提示信息， input(“请用户输入圆的半径r：”)。

• input()得到的数据类型是字符串，若要参与算术运算，需要将其转换为数值。

int( )能将数据转换为整数

float( )能将数据转换为浮点数 r=input()

eval( )能将数据转换为对应数值 r=int(r) 等同于 r=int(input())

②输出数据print(表达式)

• 在屏幕上显示文本。

• 圆括号内可以是数学表达式（计算结果）或者字符串表达式（原样输出）。

• print(表达式1,表达式2,……)可以一行输出多个数据，其输出结果会用空格隔开。

• print(字符串表达式1+字符串表达式2+……)可以一行输出多个数据，其输出结果没有间隔，相当于拼接字符串。

• print( ,end=” ”)表示设置输出打印的结束符号为空格，下一次print的内容将会直接从空格的右边开始。若end=””表示不换行。

图形用户界面, 文本, 应用程序

描述已自动生成 图形用户界面, 文本, 应用程序

描述已自动生成

1. 变量：存储一个数据

• 赋值语句：<变量>=<表达式>，此时“=”表示赋值。

• 命名规则：python变量名可以由大小写字母、数字组成，不能以数字开头，不能有特殊字符（下划线 \_ 除外），不能使用保留字，字符区分大小写。

△注意：保留字，也就是关键字，指python中的关键操作命令，if、else、print、while、 input、def、from、import等等这些都是。

• 变量的值在程序执行过程中可以通过赋值语句被改变。变量本身没有固定的类型，变量的类型是由赋值表达式值类型决定的。

|  |  |  |
| --- | --- | --- |
| **代码** | **等价代码** | **说明** |
| a=13 |  | 定义变量a，数字13赋给变量a，a的类型为int |
| a=“456” |  | 定义变量a，字符串“456”赋给变量a，a的类型为str |
| a=b=c=1 | a=1  b=1  c=1 | 定义变量a,b,c，数字1赋给变量a，变量b，变量c |
| a,b=1,2 | a=1  b=2 | 定义变量a,b，数字1赋给变量a，数字2赋给变量b |
| a=1;b=2 | a=1  b=2 | 定义变量a,b，数字1赋给变量a，数字2赋给变量b |
| a,b=b,a | a=a+b b=a-b a=a-b | a,b交换变量的值 |

1. Python 的基本运算包括算术运算、关系运算和逻辑运算三大类。

①算术运算符与算术表达式

|  |  |  |  |
| --- | --- | --- | --- |
| **运算符** | **描述** | **案例（a=10,b=20）** | **优先级** |
| （ ） | 圆括号 |  | 1 |
| \*\* | 幂 | a\*\*b返回结果10**20** | 2 |
| \* | 乘 | a\*b返回结果200 | 3 |
| / | 除 | a/b返回结果0.5（浮点数） |
| // | 整除 | a//b返回结果0（整数） |
| % | 取模（返回余数） | a%b返回结果10 |
| + | 加 | a+b返回结果30 | 4 |
| - | 减 | a-b返回结果-10 |

△注意：做算术运算时，先计算优先级高的，同优先级的按照自左向右的顺序执行。

△注意：不同类型的数据进行运算时，会发生隐式类型转换，规则是：低类型向高类型转换，int<float。例如：9\*1.0=9.0

△注意：算术运算符可以与赋值运算符“=”组合在一起，构成复合赋值运算符。+=,-=,\*=，/=，//=，%=。

|  |  |  |
| --- | --- | --- |
| **运算符** | **案例** | **等同于** |
| +=（自增） | i+=1 | i=i+1 |
| -=（自减） | i-=1 | i=i-1 |

②关系运算符与关系表达式

|  |  |  |  |
| --- | --- | --- | --- |
| **运算符** | **描述** | **案例** | **返回bool值** |
| == | 等于 | 5==8 | False |
| != | 不等于 | 4!=4 | False |
| > | 大于 | 5>4 | True |
| < | 小于 | 6<7 | True |
| >= | 大于或等于 | 9>=9 | True |
| <= | 小于或等于 | 12<=8 | False |
| 允许链式比较：x<y<z 相当于判断x<y and y<z  x<y>z相当于判断x<y and y>z | | | |
| 成员运算符：in 和 not in，用来检查一个值是否包含在指定的序列中。  “e” in “hello” —>返回True  “e” not in “hello” —>返回False | | | |

△注意：以上运算符也可以用于字符串的比较，本质上是在依次比较ASCII码值,例如“a”>“b”返回bool值False。

③逻辑运算：与或非

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **运算符** | **描述** | **表达式** | **案例** | **优先级** |
| not | 非(取反) | not a | not False 结果为True | 1 |
| and | 与 | a and b | True and False 结果为False | 2 |
| or | 或 | a or b | True or False 结果为True | 3 |

运算符的优先级：圆括号>算术运算>关系运算>逻辑运算

1. 基本数据结构
   1. 字符串（String）：字符串中的元素都可以通过索引来定位。

形状

描述已自动生成 形状

描述已自动生成 图示

中度可信度描述已自动生成

• len()返回字符串的长度，即元素的个数

• 字符串[索引号]访问个别元素

• 字符串[start:end:step]访问多个元素

**假设有字符串 s=”满分计划是由陶小波编写的”**

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成 图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

②列表（list）：一组数据序列，元素中的每个元素可以是不同类型的数据。与字符串类似，列表中的元素可以通过索引进行访问。

• 列表名.append(值)在列表末端追加新元素

• 列表[索引号]=值 能更新列表中的元素

**假设有一个列表ls=["满分计划",1234567,True]**

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

③字典（dictionary）：字典中的每个元素包含两部分内容：键和值。键和值在字典中以成对的形式出现，并以如下方式标记：d={keyl :valuel,key2:value2,…｝。字典中的元素是没有顺序的，引用元素时以键为索引。

**假设有一个字典a = {"姓名":"张三", "年龄": 20, "体重": 72, "身高": 172}**

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

**3.2.2Python控制结构**

1. 分支结构的程序实现

①单分支语句：条件判断为真时执行语句块

|  |  |
| --- | --- |
| if 条件**：**  □□□□语句块 | **案例**  x=1;y=2  if x>y:  print(“较大值为”,x) |

②双分支语句：条件判断为真时执行语句块1，条件判断为假时执行语句块2

|  |  |
| --- | --- |
| if 条件**：**  □□□□语句块1  else**：**  □□□□语句块2 | **案例**  x=1;y=2  if x>y:  print(“较大值为”,x)  else:  print(“较大值为”,y) |

△注意：通过缩进，Python能够识别出语句是隶属于if（else）。

③多分支语句：有多种情况需要判断，当判断条件2时即默认条件1为假，执行到条件n-1时即默认前面所有条件为假

|  |  |
| --- | --- |
| if 条件1**：**  □□□□语句块1  elif 条件2**：**  □□□□语句块2  **……**  elif 条件n-1**：**  □□□□语句块n-1  else**：**  □□□□语句块n | **案例**  bmi=w/h\*\*2  if bmi<=18.5:  print(“过轻”)  elif bmi<=24:  print(“正常”)  elif bmi<=28:  print(“超重”)  else:  print(“肥胖”) |

1. 循环结构的程序实现
   1. for语句：控制循环次数

|  |  |
| --- | --- |
| for 变量 in 序列 **:**  □□□□循环体 | **案例一：遍历数字序列**  for i in range(6):  print(i)  **案例二：遍历字符串元素**  s=“hello”  for i in s:  print(i)  **案例三：遍历字符串下标**  s=“hello”  for i in range(len(s)):  print(s[i]) |

* 1. while语句：控制循环条件

|  |  |
| --- | --- |
| while 条件 **:**  □□□□循环体 | **案例：**  i=1  while i<=14:  print(“居家隔离”,i)  i+=1 |

③break和continue的使用

写在循环内部，当满足某个条件时，控制循环停止，break终止整个循环，continue停止本次循环。

文本

描述已自动生成 文本, 应用程序, 聊天或短信

描述已自动生成

**3.2.3函数与模块**

1. 内建函数（Python内部实现的函数，可以直接使用）

表格

描述已自动生成

1. 自定义函数

|  |  |
| --- | --- |
| def 函数名(参数集合):  <函数体>  [return 函数值] | 案例：  def add(a,b):  c=a+b  return c |
| 调用函数：  函数名（实参） | a=add(2,3)  #此时变量a的值为函数的返回值5 |

1. 模块

|  |  |  |
| --- | --- | --- |
| **导入方法一** | **导入方法二** | **导入方法三** |
| import <模块名>  图形用户界面, 文本, 应用程序, 聊天或短信  描述已自动生成 | from <模块名> import <函数名>  图形用户界面, 文本, 应用程序  描述已自动生成 | from <模块名> import \*  图形用户界面, 应用程序  描述已自动生成 |

**math模块和random模块常用方法表（见课本）**

**3.3简单算法及程序实现**

**3.3.1解析算法**

通过分析问题的已知条件和求解目标，抽象成数学模型，借助解析式，用已知条件为变量赋值进行求解。

**3.3.2枚举算法**

按照问题要求确定问题解的大致范围，然后在此范围内对这些解进行一一列举，再对当前列举出的解进行是否满足问题要求的判断，最后把符合要求的解输出直到所有可能的解全部列举完毕。

for i in range(初值，终值，步长):

if 条件表达式 ：

输出答案/统计数

# 第四章 数据处理及应用

**4.1常见表格数据的处理**

1. Excel基础

EXCEL 是 MICROSOFT OFFICE中的电子表格程序。使用 EXCEL，可以创建工作簿（电子表格集合）并设置工作簿格式，以便分析数据和做出更明智的业务决策。

图示

描述已自动生成 图形用户界面, 应用程序, 表格, Excel

描述已自动生成

①数据区域：  
连续的区域：A5:D8 冒号隔开 不连续区域：A2:D2,H2 逗号隔开

②设置单元格格式

图形用户界面, 文本, 应用程序, 电子邮件

描述已自动生成

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 文本 | 数值 | 百分比 | 货币 | 科学计数法 |
| 00012.23 | 12.23 | 1223.00% | ￥12.23 | 12.22E+01 |

1. 数据整理

①常见数据问题及其解决方案

* 数据缺失 → 采用平均值、中间值或概率统计值来填充缺失值或忽略
* 数据重复 → 进行合并或删除处理
* 数据异常 → 对数据进行判别与剔除
* 逻辑错误 → 对与实际不符，不符合逻辑数据判别和修改
* 格式不一致 → 对数据进行转换

②数据整理的目的

检测和修正错漏的数据，整合数据资源，规整格式、提高数据质量。

1. 数据计算

在EXCEL中可通过引用函数、编辑公式等方法对工作表中的数据进行计算。计算必须先有”=“。

①公式法

公式是用户自己定义的数学表达式。常见的算术运算符有^、%、/、+、-等。

图形用户界面, 应用程序

描述已自动生成

②函数法

函数是预定义的公式，可用于执行简单或复杂的计算。常见的函数语法有=SUM（num1,num2,…）/ =AVERAGE（num1,num2,…）/ =MIN（num1,num2,…）/ =MAX（num1,num2,…）等。

图形用户界面, 文本

描述已自动生成

③自动填充

选择某一单元格，鼠标定位于右下角会出现填充柄（黑色十字），拖拽填充柄能实现对后续单元格公式的自动填充，相当于对该单元格公式的复制粘贴，且公式中涉及的数据区域也随之移动。

④单元格引用

* 相对引用：在复制/移动单元格时，单元格中公式引用的单元格会随着所在单元格位置的改变而改变。B6
* 绝对引用：在复制/移动单元格时，单元格的公式中引用的单元格不发生变化。$B$6
* 混合引用：具有绝对列和相对行或绝对行和相对列。B$6或者$B6

⑤常见错误类型

|  |  |
| --- | --- |
| #### | 列宽不够 |
| Name! | 输入函数名称错误 |
| Div/0 | 分母为0错误 |
| Value! | 引用的参数或单元格错误 |
| Ref! | 删除了被公式引用的单元格 |

1. 数据可视化

①设置图表

步骤1：选择数据区域（不连续区域用ctrl）。

步骤2：工具栏“插入”，选择图表类型（柱形图、饼图、折线图、散点图、气泡图等）

步骤3：设置图表样式，美化图表

②根据已有图表推理数据区域

看x轴、y轴和图例，讲究对称性。

图表

低可信度描述已自动生成

以上图表的数据区域为 B3**,**D3**,**B6**:**B8**,**D6**:**D8 。

**4.2编程处理数据**

1. 大数据处理思想与基本框架

# 第五章 人工智能及应用