

1 □ 第一章 基础知识

1. 转义字符(p11)
2. 重要且容易遗忘的运算符(p11-p12)
 - ① /
 - ② //
 - ③ %
 - ④ **
3. 常用内置函数 (p14-16)

abs eval float int pow len ord chr str
4. py脚本的__name__属性 import和直接执行时的不同效果

2 □ 第二章 序列

1. 明确几种常见序列结构（列表、元组、字典、集合）的基本性质（是否可写？是否有序？）
2. 序列的地址和序列元素的地址不是同一个概念
3. 列表
 - ① 支持不同数据类型的数据同时存在在一个列表中
 - ② 2维列表的概念
 - ③ append只能针对一个已经存在的列表才能进行操作
 - ④ 列表乘法的特殊现象 (p35)
 - ⑤ 列表删除元素的操作一定要从尾到头这个方向来进行处理，以防漏删
 - ⑥ 切片
 - a[2::] a[::]
 - 切片将获得一个独立的新列表
 - 对列表切片进行赋值的特殊功能

3 □ 第二章 序列

4. 元组
 - ① 序列解包
5. 字典
 - ① 对字典元素进行赋值的两种理解
 - ② get()方法
 - ③ items()方法 keys()方法values()方法
 - ④ 通过字典进行数据统计的流程一定要牢牢掌握
6. 集合
 - ① 在去重操作过程中的重要作用
 - ② 集合的几种常见计算

4 □ 第二章 序列

7. 排序

- ① key参数
- ② reverse参数
- ③ lambda的正确理解
- ④ Itemgetter在多要素综合排序过程中的重要作用 (p60~p61)
- ⑤ 为什么元组没有sort()方法?

5 □ 第二章 序列

8. 序列（特别是可写的列表、字典和集合）作为函数参数时的特殊性

- ① 函数内修改了序列参数
- ② 函数内修改了序列元素的值

6 □ 第三章 选择与循环

1. 条件表达式

- ① 对非布尔类型值进行判断时的正确理解
 - 列表是否为空?
 - None?
 - 数值为0或非0?

② 支持连续不等式

2. 选择结构

- ① 多分支选择结构 (p77)

3. 循环结构

- ① else分支最容易被忘记
- ② break和continue
- ③ while True是很常见的代码结构

4. 质数的判断问题

5. 闰年的判断问题

7 □ 第四章 字符串

1. 字符串是不可变序列结构

2. 字符串也支持切片，但不支持对切片进行赋值

3. 字符串格式化

- ① format别忘了 `print("{:4d}".format(1234567))`是什么意思?
- ② 建议大家还是使用%的方法吧
- ③ 如何控制填充字符?
- ④ 左对齐和右对齐的控制，默认是右对齐
- ⑤ %s可能会被忘记
- ⑥ %m.nf

4. 最常用的字符串函数: split,strip,join,eval,int,float

8 第四章 字符串

5. 正则表达式

- ① \在正则表达式中的作用机制：绝大多数是元字符的前缀标志，所以很多时候要防止其发生转义
- ② 正则表达式的理解步骤：
 - 字符串转义：该转义就转义
 - 正则转义：识别元字符和普通字符
- ③ 常用的正则表达式
 - `r\d+`：查找整数
 - `r\w+`：查找由数字、字母和下划线组成的单词
 - `r'[A-Za-z]+'`：查找纯粹由英语字母所构成的单词
 - `r'ba\w+a\b'`：查找由a开头和由a结尾的单词
 - `r'b\w*a*\w\b'`：查找包含a字符的单词
 - 域名字符串的查找操作（p105第6条）

9 第四章 字符串

5. 正则表达式

- ④ `findall`, `search`和`match`的不同功能
- ⑤ 对成组和子模式的正确理解
- ⑥ 如何通过`search`来查找所有特征子串？（p111-p112）
- ⑦ `match`和`search`方法中的`group`, `start`, `end`, `span`, `groups`函数的使用

10 第五章 函数

- 1. 位置参数和关键参数
- 2. 参数默认值
- 3. 可变长度参数
- 4. 实参和形参之间的参数传递关系，密切关注、慎用并尽量不用“双向传递”
- 5. `map`函数，`reduce`函数，`filter`函数（p134-p135）

11 第七章 文件

1. 文件打开

- ① 文件路径问题，别忘了绝对路径的字符串表示方法，而且期末考试肯定将使用绝对路径
- ② `r`和`r+`的区别
- ③ `w`和`w+`的区别
- ④ 文件的关闭问题
- ⑤ `with`结构的使用

2. 文本文件的操作方法

- ① `read`
- ② `write`

- ③ readline
- ④ readlines
- ⑤ writelines
- ⑥ 换行符要自己处理

12 常见程序结构

1. `[int(i) for i in str(num)]`
2. 单词或字符出现次数的统计
3. 大小写不敏感和敏感
4. 按照数的某些位数字进行排序
5. 排序时无法直接在lambda表达式中表示排序条件时的处理策略：定义一个函数来专门计算此排序要素
6. 素数判断，水仙花数判断，回文数判断
7. `a=list(set(a))`
8. 2维矩阵的相关操作：转置、相乘
9. 如何获取1个列表中可能存在的多个相等的最大值以及这些最大值元素出现的位置？