

# 实验七 Python 字符串处理

## 实验目的

- (1) 了解 6174 猜想
- (2) 熟练使用程序控制结构
- (3) 了解标准库 `itertools` 中 `permutations()` 函数的使用
- (4) 熟练使用字符串的处理方法
- (5) 分词模块的使用

## 实验内容和要求

### 1. 验证 6174 猜想

1955 年，卡普耶尔对 4 位数字进行研究，发现了一个规律：对任意各位数字不相同的四位数，使用该数各位数字能组成的最大数减去能组成的最小数，对得到的差重复进行以上操作，最终总能得到 6174 这个数字，并且操作过程不超过 7 此。

编写程序，使用枚举法（即对所有符合上述条件的四位数字）对这个过程进行验证，并打印每个数对应的操作次数。

提示：`permutations` 函数使用方法：`permutations(iterable, r)`

该函数接受两个参数，`iterable` 为一个可迭代的序列结构，`r` 为整数，函数返回值为 `iterable` 结构中 `r` 个元素的排列序列。

### 2. 简单垃圾邮件识别

- a) 利用文本模拟垃圾邮件内容。设定常见垃圾邮件特征词汇，如“发票”、“购买”、“征稿”、“促销”等，并设定每个特征值对应的垃圾邮件概率  $p_i$ ，利用公式  $P=1-(1-p_1)*(1-p_2)*\dots*(1-p_i)$  来计算邮件是垃圾邮件的概率，设定阈值，对垃圾邮件进行判定。尽量提高准确度，减少漏盘和误判。
- b) 反干扰措施：垃圾邮件发送者往往会在邮件中插入一些特殊符号如【、】、-、\*、/等，干扰对邮件特征词的提取，但一般邮件中以上特殊符号出现的总量应该不会太多，如果此类符号出现的总量超过一定阈值（自己设定），不需要进行 a) 步骤的判断，即可直接判定为垃圾邮件。

编写函数，实现垃圾邮件判定器，并构造数据进行验证。