



Python练习题

入门级别

1. 【Q01】求用十进制，二进制，八进制表示都是回文数的所有数字中，大于十进制20的最小值。
2. 【Q02】数列的四则运算

在一串数字中加入适当的运算符，使其结果和是原来的数字的翻转，同时要遵循四则运算的顺序进行（先乘除，后加减。）某些位数之间可以没有运算符，但至少插入一个运算符。

例如：在100~999，符合条件的有以下几种情况：

$$351 \rightarrow 3 * 51 = 153$$

$$621 \rightarrow 6 * 21 = 126$$

$$886 \rightarrow 8 * 86 = 688$$

问题：求位于1000~9999，满足上述条件的数。

解题思路：

要将一个4位数分割开，并在其中添加运算符，然后对该算式进行运算。这里需要用到Python的eval()函数。

同时考虑到不符合运算规则的算式，如：1+0+01,1 / 0 + 11等

```
def Q02():
```

```

oplist = ['+', '-', '*', '//', ''] # 这里用到了//是Python中的整除运算, ''为没有运算符。
for i in range(1000,10000): # 从1000开始循环到9999
    num_str = str(i) # 将i转换成字符串, 方便后面的运算
    for op1 in oplist: # 第一个运算符
        for op2 in oplist: # 第二个运算符
            for op3 in oplist: # 第三个运算符
                val = num_str[0] + op1 + num_str[1] + op2 + num_str[2] + op3 +
num_str[3]

                try:
                    temp = str(eval(val)) # temp为拼接后的字符串的运算结果, 需要转换成
字符串, 以便后面翻转。
                    if num_str == temp[::-1] and len(val) >4: # 因为题目要求至少有一个运算符
                        print("符合条件的数字为: {},表达式为{},其运算结果为:
{}".format(num_str,val,temp))
                except:
                    pass # 对算式不正确的不进行任何处理

if __name__ == "__main__":Q02()

```

这里用的比较暴力的办法, 其实分析题目后可以得到, 除了用乘法, 其他的运算都无法达到目的。
用加法最大的值为: $999+9 = 1008$, 反转时不可能得到原始值的。