Algorithm Puzzles解题笔记

目录

[Algorithm Puzzles解题笔记 1](#_Toc129092073)

[1. 概要 1](#_Toc129092074)

[2. Q02: 四则运算组合游戏 2](#_Toc129092075)

[2.1 问题描述 2](#_Toc129092076)

[2.2 解题分析 2](#_Toc129092077)

[2.3 算式值的计算 3](#_Toc129092078)

[3. Q04: 切分木棒 4](#_Toc129092079)

[3.1 问题描述 4](#_Toc129092080)

[3.2 数学解法 4](#_Toc129092081)

[4. Q05: 硬币兑换 5](#_Toc129092082)

[4.1 问题描述 5](#_Toc129092083)

[4.2 递推表达式 5](#_Toc129092084)

[5. Q07: 日期的二进制转换 5](#_Toc129092085)

[5.1 解题分析 6](#_Toc129092086)

[5.2 优化 6](#_Toc129092087)

[6. Q08: 优秀的扫地机器人 6](#_Toc129092088)

[6.1 问题描述 6](#_Toc129092089)

[6.2 解题分析 7](#_Toc129092090)

**[Notations and Abbreviations]**

# 概要



本文是关于该书的完整的python解题笔记。

# Q02: 四则运算组合游戏

## 问题描述

在任意4个数字（0,1,2,…,9）构成的数字序列每两个数字之间中插入四则运算符{+, -, \*, /}或者不插入，计算由此得到该算式的结果。

例1：’1+2-3\*4’ = -9

例2：’1+23\*4’ = 93, 此例中，2与3之间没有插入运算符，因此构成一个两位数

如果要求计算结果等于原数字序列逆序排列构成的四位数，求满足以上条件的数。

但是要求至少要插入一个运算符。因为一个都不插入的话，其实得到的就是原4个数字表示的四位数，那任何一个四位回文数都满足这个条件，这个结果没有什么意义。

为了避免一些trivial solution，把10的倍数都排除在外，这样确保算式评估结果以及逆序数仍然为有效的四位数。

## 解题分析

本题涉及两个要点：

[算式的构成]

算式的构成就是构造不同的将运算符插入数字之间的组合，可以把空字符（注意，不是说white-space，而是说长度为0的字符串“”）也看成是一个运算符，插入空字符也就相当于是不插入。

这样总共就有5个运算符：{‘+’, ‘-’, ‘\*’, ‘/’, ‘’}，而四个数字构成的序列中共有3个位置可以插入，再扣除不能插入三个空字符‘’，因此总共有5^3-1=124种组合。

在本体中考虑用字符串的方式表示所得到的算式，比如说：‘1+2-3\*4’.

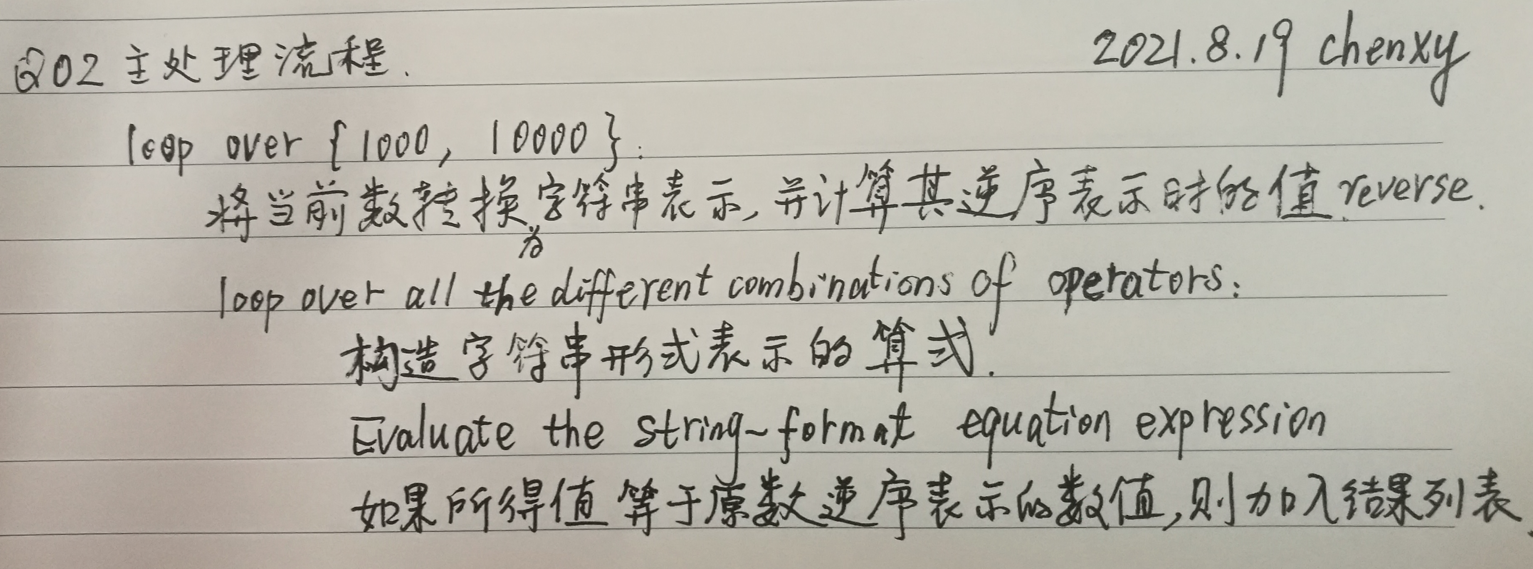
构造好算式后，接下来就是计算该算式的值。

[算式值的计算]

以上第一步中所构造的算式是所谓的中序表达式，而且没有括号，属于相对简单的情况。计算机中对算式的估计通常是将算式先由中序表达式转换为逆波兰表达式，然后再对逆波兰表达式进行评估处理。

  感觉这还是一个比较有意思的东西，具体算法思路将另文单独解说。

基于以上讨论，本题的处理流程如下所示：



## 算式值的计算

To be added.

# Q04: 切分木棒

## 问题描述

假设要把长度为n厘米的木棒切分为1厘米长的小段，但是每段木棒只能由1人切分。比如说，当木棒切分为3段，可以由3人同时分别切分各段。求最多m个人时，最少要经过几轮（原文是几次，容易误解）才能完成切割。

比如说n=8, m=3时，分4轮切分即可。第1轮，由1个人将原始木棒切分为两段；第2轮（此时有2段木棒），由2个人同时它们切分为共4段；第3轮（此时有4段木棒），由3人分别对其中3段进行切分操作；第4轮（此时还剩1段木棒需要切分），由2人将最后剩下的1段切分为2段。

## 数学解法

这道题目其实数学的方式来解决可能更为简单。“数学解法”是相对于“编程解法”的说法，即仅通过数学或者逻辑的推到的方式求解的方法。以下我们通过数学推导的方式得到本题的解析解或者说闭式解(closed-form solution)

首先，将n厘米的木棒切分为1厘米长的小段总共需要（n-1）次切分

其次，在有m个人时，每一轮最多可以（由m个人同时工作）进行m次切分操作。当木棒段数还小于m时，当前轮次所能进行的切分操作数受限于木棒段数；当木棒段数大于等于m时，当前轮次所能进行的切分操作数为m

第三，在第1轮过后，木棒数变为2段；在第2轮过后，木棒数变为4段；在第k轮过后，木棒数变为2^k(2的k次方)段。因此到第轮时，因为所能执行的切分操作数仍然是受限于木棒段数的。从第轮开始，则可以每轮最多执行m次切分操作

基于以上观察，可以知道：

1. 在前轮中总共会将木棒切分为段，总共进行了次切分操作
2. 接下来还需要的切分操作数为，由于每轮可以执行m次切分操作，因此还需要轮

因此，总共需要轮。

当n = 8, m = 3时，代入上式可得答案为4

当n =20, m = 3时，代入上式可得答案为8

当n =100, m = 5时，代入上式可得答案为22

# Q05: 硬币兑换

## 问题描述

公交车上的零钱兑换机可以将纸币兑换成10日元、50日元、100日元和500日元的硬币，且每种硬币的数量都足够多。因为兑换出太多的硬币不太方便，只允许机器兑换成最多15枚硬币。比如说1000日元的纸币就不能对换成100枚10日元的硬币。

求兑换1000日元纸币会出现多少种不同组合？注意不计硬币兑出的先后顺序（即可以认为硬币是一起出来的）。

## 递推表达式

这道题目可以表达为如下数学方程的形式：

这是一道线性规划（Linear Programming）问题。

More generally, 令所需要兑换的纸币记为money，可用的硬币以数组形式（降序排列）表示为coins，最多允许的硬币数为maxNum。记f(money, coins, maxNum)表示符合题设要求的组合数。

首先考虑最大面值的硬币coins[0]（假定coins中按降序排列）的使用。很显然，coins[0]最少可以用0枚（即不同），最多可以用枚，据此可以把问题分解为若干个更“小”的子问题来求解。由此可以得到以下递推关系式：

Baseline cases或者说边界条件如下：

# Q07: 日期的二进制转换

不同国家的日期的表示方式（习惯）不相同。

把年月日表示为YYYYMMDD这样的8位整数，然后把这个整数转换成二进制数并逆序排列，再把得到的二进制数转换成十进制数，求与原日期一致的日期。求得的日期要在上一次东京奥运会（1964-10-10）到下一次东京奥运会（2020-07-24）之间。

## 解题分析

其实，就是找转换为二进制数后构成了回文数（关于回文数参见Q01）的日期。

本题的焦点之一是关于日期的处理，纯粹人工处理的话需要考虑月份大小、闰年以及二月份这种特殊月份等情况，非常容易出错。好在各种编程语言都有内置的库来做这件事情。Python中由datetime模块来处理，所以问题变成了datetime模块的使用的问题。特别是日期递增的处理方式。

焦点之二是字符串的处理。

在以下代码中，有以下几个细节值得注意（不一定是最优的，只是本渣采用了这样比较笨拙的办法而已^-^）：

1. 由date变成string后中间会有分隔符“-”（没有找到不包含“-”的字符串变换方式），需要去除。这里用str.replace()处理
2. 十进制数变成二进制数时输出的字符串，头上以“0b”开始，这个在进行是否回文数的判断中需要先去掉
3. 回文数的判断，这个采用和Q01相同的处理方式，用str[:,:,-1]的方式取字符串逆序再与原字符串比较

## 优化

原书中提示了可以利用构成回文数的日期的特性可以大幅度削减搜索范围。当然其代价就是解决方案的可读性以及可扩展性。

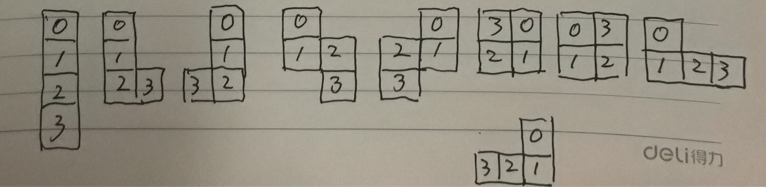
To be discussed.

# Q08: 优秀的扫地机器人

## 问题描述

现在市面上有些扫地机器人有时候会反复地清扫某一个地方。

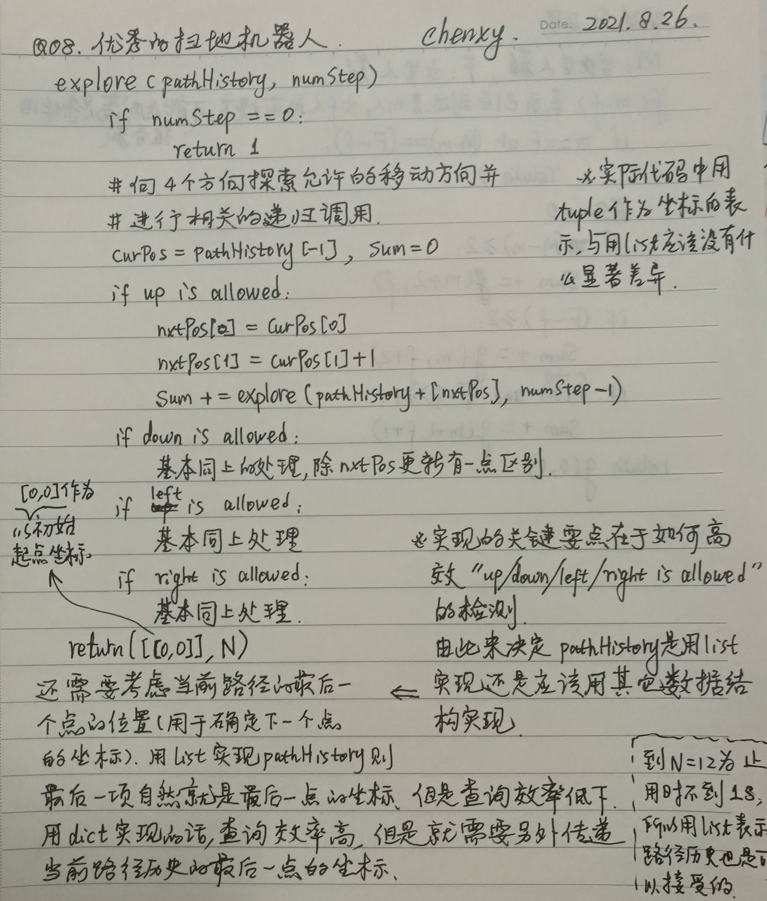
假设有不会反复地清扫某一个地方的扫地机器人，它只能前后左右移动。举个例子，如果第1次向后移动，那么连续移动三次后，它的轨迹会出现9种情况，如下所示(0表示起点的位置，k={1,2,3}表示经过k步移动后机器人所在的位置)。



求这个机器移动12次时，有多少种移动路径？

## 解题分析

深度优先路径遍历搜索问题。



[Reference]

[Revision history]