

1、问题描述

给定一块电路板，分为上下两部分，上有n个接线柱，下有m个接线柱，要求使用导线将上下接线柱进行连接，且导线不相交，求解最大的导线数量或者是上（下）线柱的最大使用数量。

记 $(i, \pi(i))$ 为上下接线柱的一条连线， $Nets(i, j)$ 为连线的集合，则有

$$Nets(i, j) = \{(i, \pi(i)), 1 \leq i \leq n\}$$

原问题转换为求解 $Nets(i, j)$ 的最大不相交子集，该问题存在有序性，也即对于上线柱中的任意两个线柱 $i, j \leq n$ ，如有序，则必然下线柱也是有序的，也即

$$1 \leq i < j \leq n \iff \pi(i) < \pi(j)$$

问题的输入为：下线柱的接线位置

如input =[1,2,3]表示，有3个上线柱，且从小号到大号依次接线到下线1，2，3线柱上。

2、分析

记 $N(i, j) = \{t | (t, \pi(t)) \in Nets, t \leq i, \pi(t) \leq j\}$ ，则 $N(i, j)$ 表示对于小于或等于给定的上下两点 i, j 之间连线方式的集合， $N(i, j)$ 的最大不相交子集记为 $MNS(i, j)$ ，

$$Size(i, j) = |MNS(i, j)|。$$

(1) 当 $i = 1$ 时

$$MNS(1, j) = N(1, j) = \begin{cases} \emptyset & \text{if } j < \pi(1) \\ \{(1, \pi(1))\} & \text{if } j \geq \pi(1) \end{cases}$$

(2) 当 $i > 1$ 时

若 $j < \pi(i)$ ，此时则有 $(i, \pi(i)) \notin N(i, j)$ ，从而有

$$MNS(i, j) = MNS(i-1, j), Size(i, j) = Size(i-1, j)$$

若 $j \geq \pi(i)$ ，则对 $\forall (t, \pi(t)) \in N(i, j) \wedge t < i \implies \pi(t) < \pi(i)$ ，在这种情况下 $N(i, j) \setminus \{(i, \pi(i))\}$ 是 $N(i-1, \pi(i-1))$ 的最大不相交子集。

若 $(i, \pi(i)) \notin MNS(i, j)$ ，则对

$$\forall (t, \pi(t)) \in MNS(i, j), \text{有 } t < i。 \text{从而 } MNS(i, j) \subseteq N(i-1, j)$$

因此 $Size(i, j) \leq Size(i-1, j)$ ，另一方面，又因为

$$MNS(i-1, j) \subseteq N(i, j) \implies Size(i, j) \geq Size(i-1, j)$$

所以 $Size(i, j) = Size(i-1, j)$

3、递归子结构

1) 当 $i = 1$ 时

$$Size(i, j) = \begin{cases} 0 & \text{if } j < \pi(1) \\ 1 & \text{if } j \geq \pi(1) \end{cases}$$

2) 当 $i > 1$ 时

$$Size(i, j) = \begin{cases} Size(i-1, j) & \text{if } j < \pi(i) \\ \max(Size(i-1, j), size(i-1, \pi(i)-1)) + 1 & \text{if } j \geq \pi(i) \end{cases}$$

4、代码

```

1 '''
2 电路布线
3 input:
4     dim=[7,6,3,1,4,0,8,2,9,5]
5 dim的含义指：上线柱从小到大在下线柱的接线位置，上下线柱的标号均从0开始，此处要区别
6 理论部分的下标是从1开始。
7     top, down = len(dim), max(dim) + 1
8 top,down用于创建记录N(i,j)的最大长度的矩阵
9 输出最佳布线方式，可以参考理论部分，最大长度的矩阵的生成方式。
10 output:
11     [(8, 9), (6, 8), (4, 4), (2, 3)]
12 '''
13 def mNS(top,down,dim=[]):
14     '''
15     :param dim:上线柱从小到大依次接到下线柱的位置，如L=[1,2,3,4,5,6]
16     :return:
17     '''
18     top, down = top,down
19     s = [[0 for i in range(down)] for i in range(top)]
20     for i in range(dim[0], down):
21         s[0][i] = 1
22     for i in range(1, top):
23         for d in range(dim[i]):

```