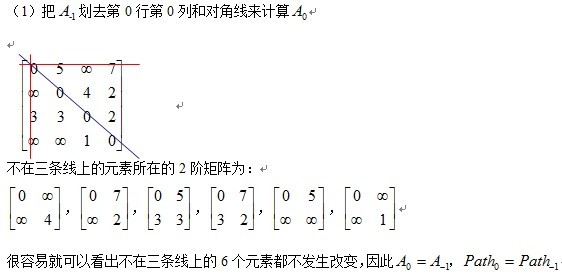
**oyd-Warshall算法过程中矩阵计算方法—十字交叉法**

转载 2014年07月29日 16:28:41

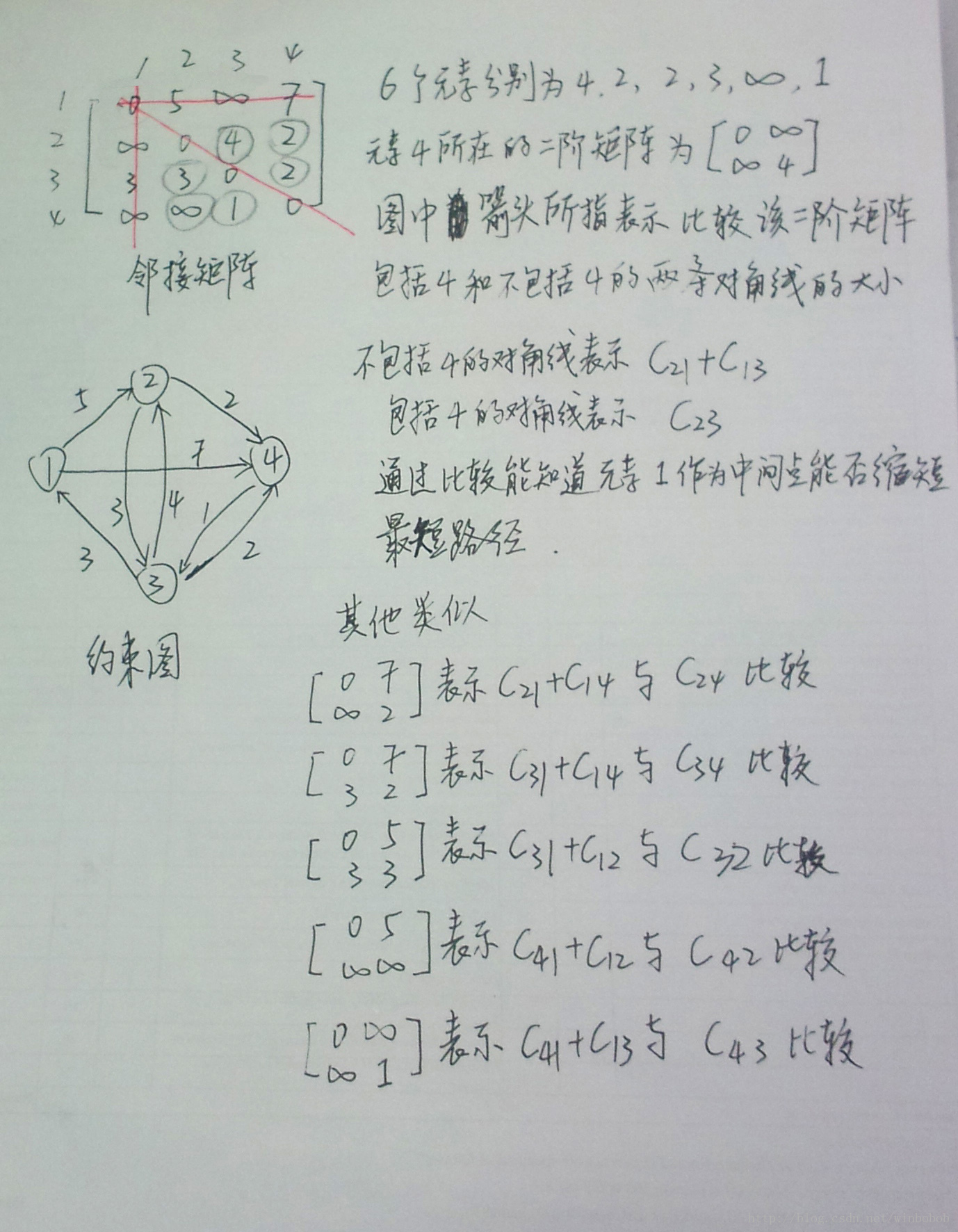
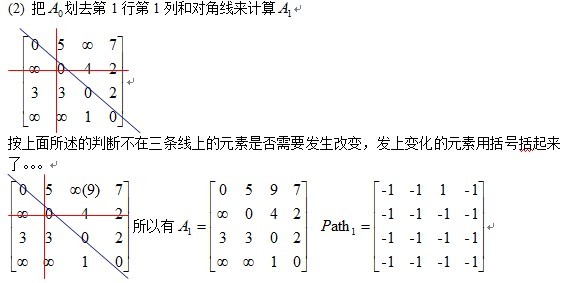
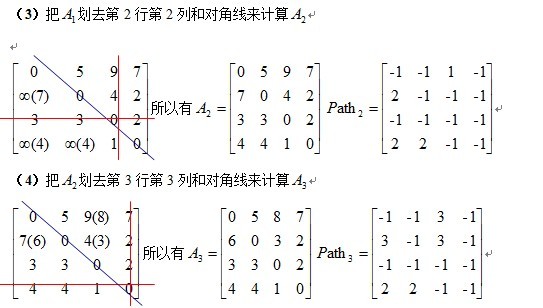
* 标签：
* [十字交叉法](http://so.csdn.net/so/search/s.do?q=%E5%8D%81%E5%AD%97%E4%BA%A4%E5%8F%89%E6%B3%95&t=blog) /
* [Floyd-Warshall](http://so.csdn.net/so/search/s.do?q=Floyd-Warshall&t=blog) /
* [矩阵计算](http://so.csdn.net/so/search/s.do?q=%E7%9F%A9%E9%98%B5%E8%AE%A1%E7%AE%97&t=blog)
* 5987

前几天在看Floyd算法的时候，虽然感觉程序很简单，但是让你动手写那些过程矩阵的时候就感觉不怎么简单了，就上网找找看有木有简便的计算方法，搜索之后没有发现有现成的例子，只搜到了两句“弄两条线，从左上角挪到右下角”，“十字交叉法，从左上角到右下角”，除此之外就再也木有找到有用的东西了。没有内容就创造内容，填补空白！

  
先来简单分析下,由于矩阵中对角线上的元素始终为0,因此以k为中间点时,从上一个矩阵到下一个矩阵变化时,矩阵的第k行,第k列和对角线上的元素是不发生改变的**（对角线上都是0，因为一个顶点到自己的距离就是0，一直不变；而当**k为中间点时，k到其他顶点（第k行）和其他顶点到k（第k列）的距离是不变的**）**。

因此每一步中我们只需要判断4\*4-3\*4+2=6个元素是否发生改变即可,也就是要判断既不在第k行第k列又不在对角线上的元素。具体计算步骤如下：以k为中间点（1）“三条线”：划去第k行，第k列，对角线（2）“十字交叉法”：对于任一个不在三条线上的元素x，均**可与另外在k行k列上的3个元素构成一个2阶矩阵**，**x是否发生改变与2阶矩阵中不包含x的那条对角线上2个元素的和有关，若二者之和小于x，则用它们的和替换x，对应的Path矩阵中的与x相对应的位置用k来替代。**。。  
http://hi.csdn.net/attachment/201109/7/0_131538987200Qq.gif  
下面来具体看高分笔记上面的那个题目吧。。。。  


详细图解：

  
  
  
  
  
经过以上4步就可以得到最终结果了。。。写的比较繁琐，但是看明白了之后，计算将会非常简单啊。。。。。