

第三次作业题目讲解

题目讲解假设大家能理解链表讲解里的内容。

A - 连续线段

首先，我们将线段表示为链表中的节点，每个节点有 x_1, y_1, x_2, y_2 四个字段（或者说起始点和结束点两个字段）。对于每个线段 p ，我们遍历所有其它的结点，如果 p 的起点与 q 的终点重合，则让 q 作为 p 的前驱；如果 p 的终点与 q 的起点重合，则让 q 作为 p 的后继。由于没有三条或三条以上的线段共一个端点，所以对某个线段不可能有两个前驱或者两个后继，即前驱后继各用一个指针域，最普通的实现就可以了。

最后，遍历每个线段 p ，看看以它为起点的最长序列的长度（遍历，直到遇到 `NULL`），这里的逻辑就类似于一个寻找最大值的操作。

B - 多项式相乘

在读入的时候可以利用下面的技巧分别读入两组数据（因为题目中特意说了最后一行结尾有一个换行符，而没有空格）：

```
while(scanf("%d%d%c", &a, &b, &ch)) {  
    // 进行结点 (a, b) 的插入  
    // ...  
    if(ch != ' '){  
        break;  
    }  
}
```

然后，我们可以利用一个双重循环，将系数相乘，指数相加的新节点 p 插入保存结果的链表，同时保持指数的有序性，如果在保存结果的链表中已经有了指数相同的结点，那就直接将 p 的系数直接加在上面就行了。

最后输出时，遇到系数为 0 的项直接跳过就行了。