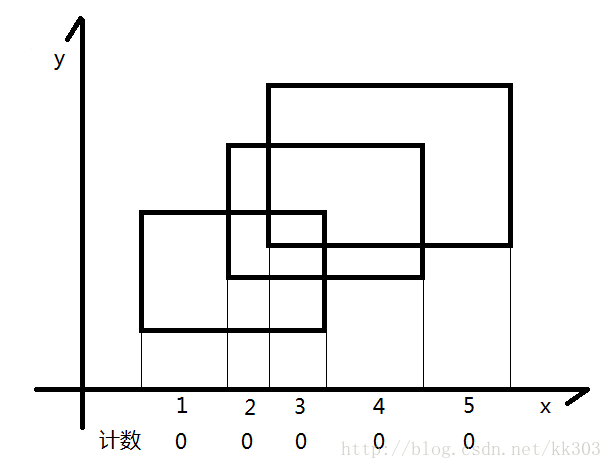
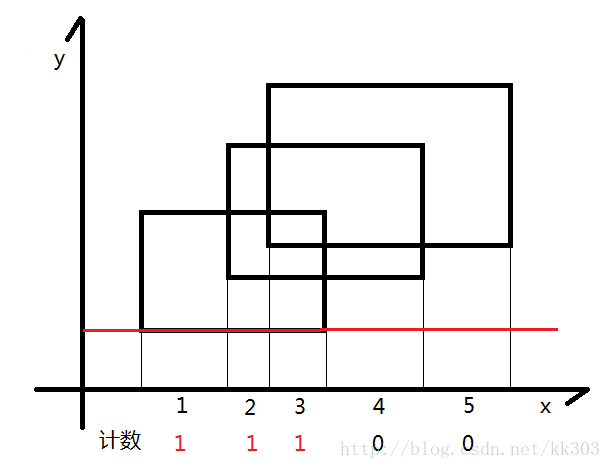
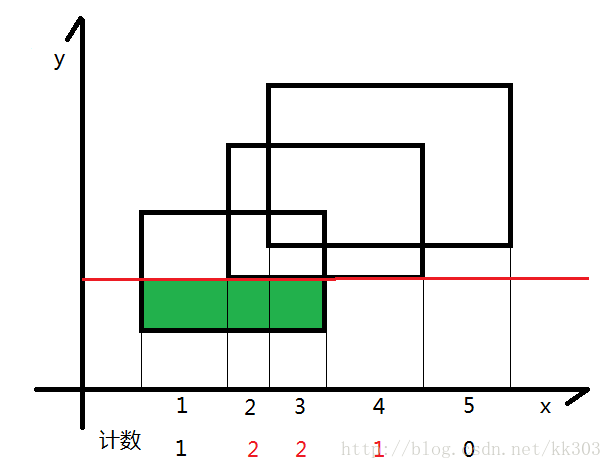
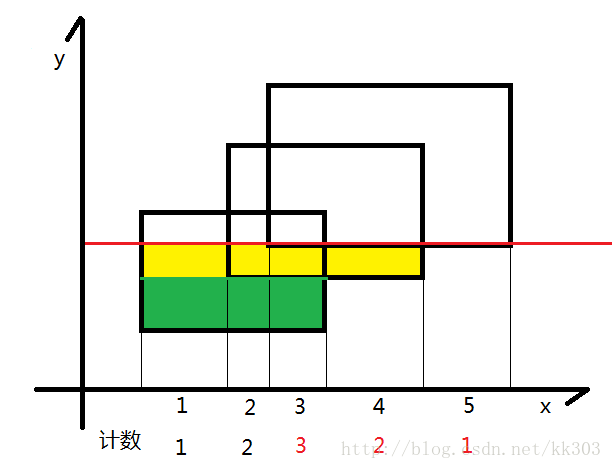
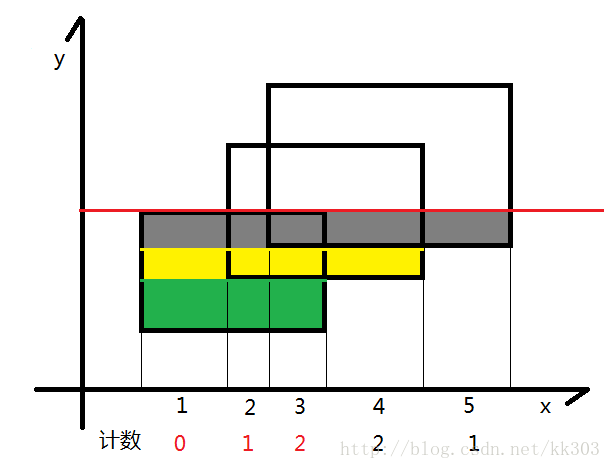
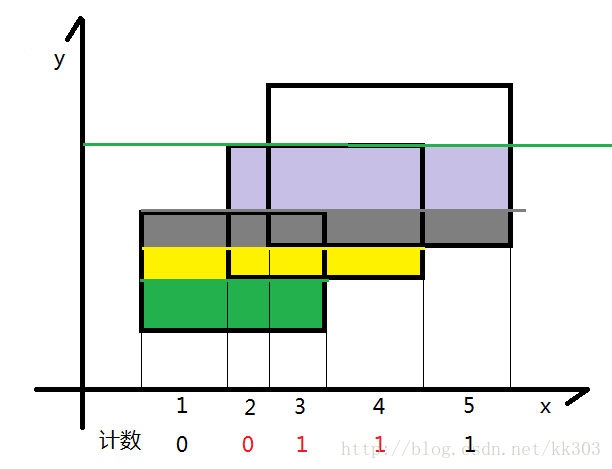
这其实算是一个基本操作吧，对于矩形求并，考虑扫描线。

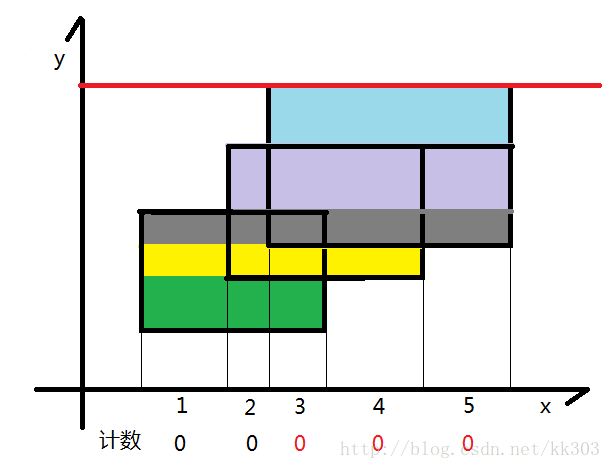
对于此矩形(下图)怎么去求面积









(图片来源于网站 [https://blog.csdn.net/riba2534/article/details/76851233#](https://blog.csdn.net/riba2534/article/details/76851233))

这就是如何去求用扫描线分层求面积，分别计算每一颜色的面积相加。

那么我们如何去确定长与宽呢，我们对于每一个矩形保留长，并且y1此行所对的行标记为+1,所对应的四元组为(x1,x2,y1,+1),而y2行所对应的四元组为(x1,x2,y2,-1)。

先进行以y升序排列以后，所对应的长为当前(x1,x2)所对应的计数所对应的长度，宽为这个与后一个y值得差。

这是一个O(n^2)的算法。但是考虑计数时记录了许多无用状态，首先考虑离散化矩形，重新编号，如上图，第一个是1号，第二个是2号，以此类推。

所以此刻就可以用线段树去维护起(x1,x2)里的长的长度以及对于(x1,x2)中所有数+w操作，复杂度为O(n\log n)

但是这个线段树因为比较特殊，只需要查询次编号有没有覆盖所以直接pushup即可，代码中好像叫做pushdown，懒得改了。