题意 : 给你一个长度为2e5的数组 ，问有多少个区间[l,r] 满足区间中元素值之和<t .

思路： 一开始想的就是先搞出前缀和，然后我们枚举右端点r 把之前经过的pre 存入map 里 ，然后利用lowbound 查合法的l 有多少个 ，但是我发现map（set）的位置是不连续的，也就是无法 用返回位置-map.begin()得到合法前缀和个数有多少个；既然这个有问题，我们再换一种思路，考虑一下**分治**能否解决这个问题，我们假设work(l,r) 表示算出[l,r]内合法区间对数，那么他怎么由work(l,mid) 和 work(mid+1,r) 维护过来呢？ 显然没有计算的合法对数 只能是左端点在[l,mid]内，而右端点在[mid+1,r]内部 ，我们考虑如何高效求出来这样的合法区间，不难想到对于work（l,r）我们需要维护出来这个区间的前缀和以及后缀和，这个维护很简单。那么我们可以对于[l,mid]的后缀和存入vector 枚举每个 右端点 再用二分速查个数

这一层的复杂度为nlogn ，总共logn层 所以复杂度为nlognlogn

ps（不过看到好多大神用的lowbit 代码还很短 日后可以学一波）