20180213小测

上了几天课后，我么迎来了秦神的考试……

上次秦神出题T1T3集体爆零的心里阴影还没有褪去QAQ……

T1:

说白了就是让你在原串中选出一些子串使之价值和最大，已经被删去的子串使得区间两端合并。

首先注意这题是子串而不是子序列。考试的时候我当子序列做了半天，什么贪心、DP、网络流全都试了一遍，发现根本不可做，连爆搜都不会写……

然而看到这题这样就肯定是区间DP了，因为我们选择的是一个子串，所以对于一个我们要选出的串，他要么在原串中连续出现，要么在原串中分开出现且间隔部分可以全部被选出。

所以我们令f[i][j][p][k]表示原串区间[i,j]，匹配目标串p，第i位匹配第1位，第j维匹配第k位，dp[i][j]表示原串[i,j]能否被完全选出(先别管我为什么这么定义，一会看转移就明白了QAQ)。

我们考虑串p的位置k-1匹配的位置。我们在区间[I,j)枚举l，表示第k-1位匹配的位置。

若l==k-1，则中间没有间隔串，f[i][j][p][k]从f[i][l][p][k-1]转移。

若l!=k-1，说明中间间隔了一些部分且他们必须完全消去，所以我们让f[i][j][p][k]从f[i][l][p][k-1]&&dp[l+1][j]转移。

对于dp数组，一个能完全选出的区间要么由两边两个能完全选出的区间组成，要么恰好完美匹配某个目标串，我们枚举一下分割位置或者匹配的目标串即可。

复杂度？f的转移把[p][k]摊还下来是O(n^4)的，且为区间DP，有一个最大1/8的常数。dp的转移只用枚举中间位置或匹配串，复杂度O(n^3)。实测跑得飞快。

然而这题限制64mb内存，你直接开会MLE。用vector进行动态resize依旧会MLE(亲测还会TLE)。你需要开出f[i][j][p]的指针，最后一维k用C++的new分配内存就好了。

关于答案统计，我们可以看做在区间内选择几条有权值且不相同的线段覆盖，做线段覆盖就好了。

代码:

T2:

关于题意:如果若干个点全都距离很近能相互到达，使得能到达每家店的数量大于等于k，则全部关门。

我们把距离转成切比雪夫距离，那么一家店能到达的店在一个矩形内。我们只用统计出一家店能被多少家到达，就可以知道他是否需要关门。

二维数据结构随便做啊，然而64mb内存卡树套树，懒得调扫描线或分治。考虑n只有10w，分块的数据范围，所以好写好调的kdtree随便过啊。

我们对于每个节点记录这个节点被完整覆盖了多少遍，最后再把树dfs一遍好了。统计答案的时候别忘了减去自身对自身的贡献。

码了十几分钟考场一发AC，话说这题为什么就我一个拿分的？别人都写跪了？

代码:

T3:

不能相互到达？不就是最大独立集？点数减最大匹配就好了。

等等，一般图最大匹配？这不是经典的NP问题吗？这题能做？

发现按照行列分组后这是一个二分图，然后就拿到了48分……

这个思路为什么有问题呢？因为一旦n为奇数的话，这图存在奇环，就不是二分图了……

根据鸽笼原理，一定有一行点数<=10，我们暴力枚举这一行状态，剩下的就是二分图了。

说着简单写着难系列。

考场48分代码：

考后AC代码：

话说在从老家回来汽车上写这东西真是晃死我了……