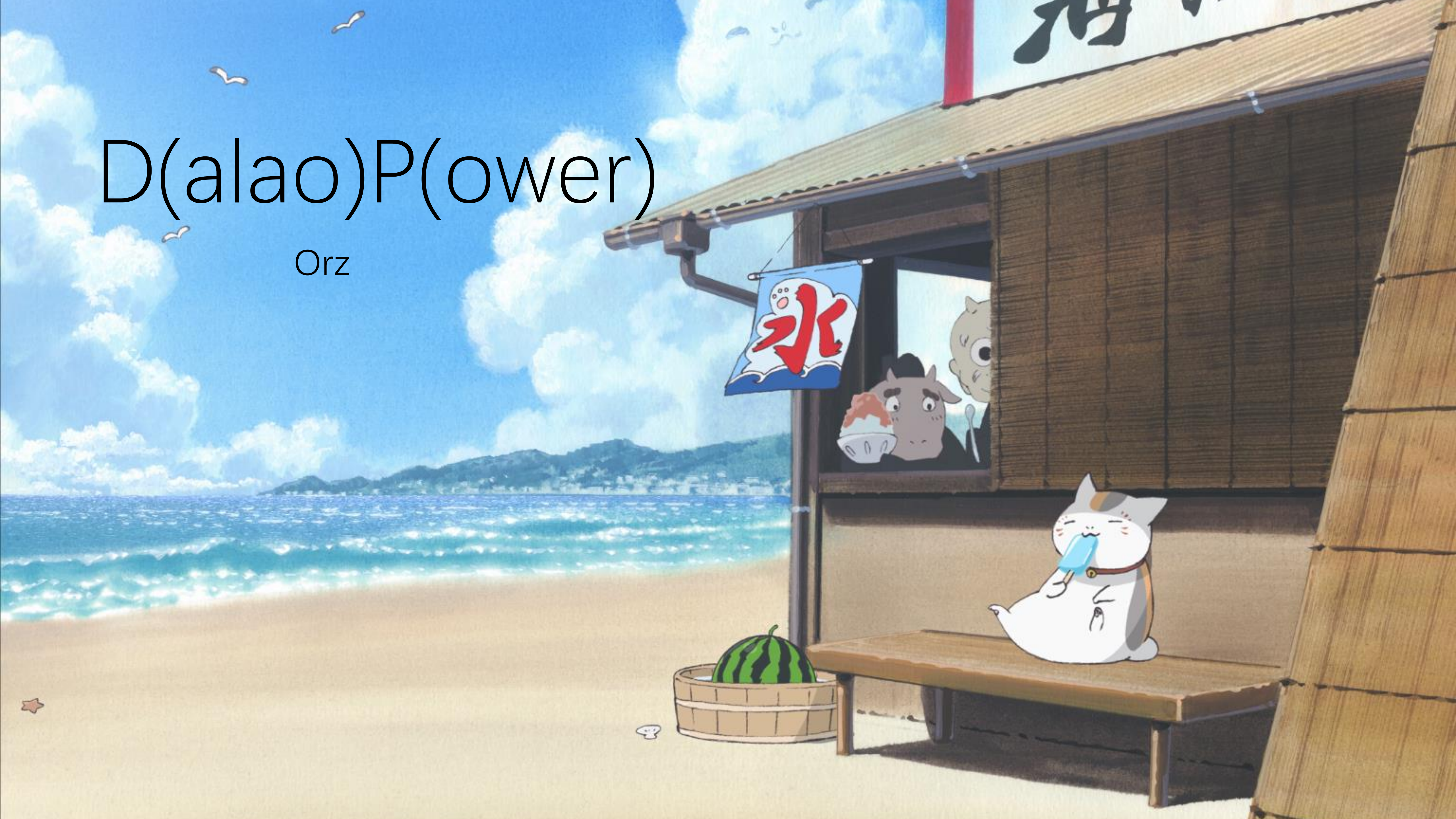
D(alao)P(ower)

Orz



Warning

- 课件里可能会出现各种锅或者歧义

- 想到题目做法的同学可以上来讲

动态规划

- 可难可易
- 有短小精悍的，也有极其繁琐的
- 套在哪里都可以

- 知识点少
- 但有各种模型需要了解。
- 足够强的话可以当场开脑洞找姿势

Outline

根据一般的dp题难点，分为三个部分

- 状态的划分
- 转移的优化
- 一些常见的dp模型

- 你可以把这个课件当作有分类的杂题选讲

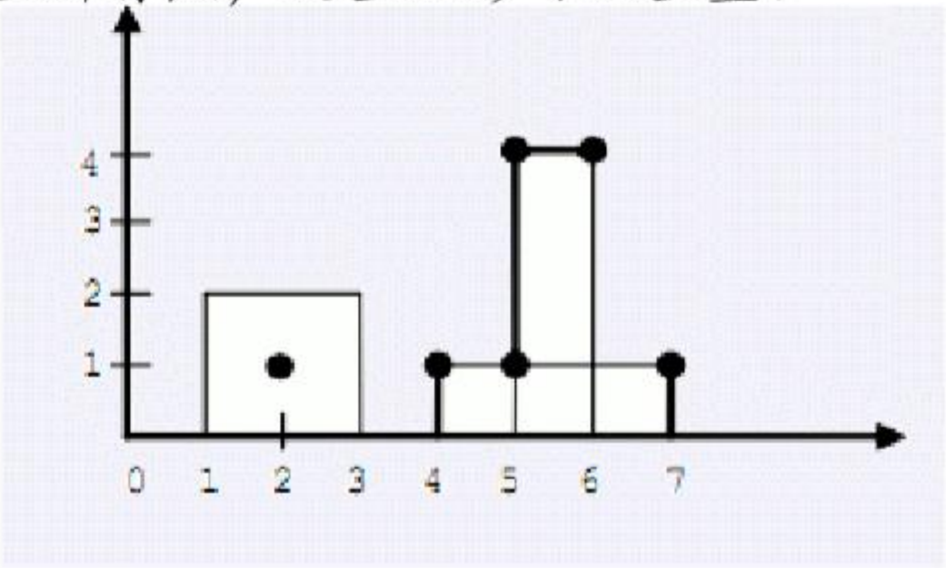
状态的划分

- 利用题目的独特性质
- 方便转移

- ~~好写就行~~

Photo

平面上有 n 个点，要求用最少的底边在 x 轴上且面积不超过 A 的矩形覆盖所有点，这些矩形可以重叠。



- $1 \le n \le 100, 1 \le A \le 200000, 0 \le x \le 3000000, 1 \le y \le A$。
- Source: CEOI 2009

sol

- 一定存在最优解，满足所有矩形x轴区间只有包含或相离，没有相交。
- 换句话说，**矩形是树形结构**，并且儿子节点高度>父亲。
- 设$F[i][j][k]$表示覆盖区间[i,j]，高度>=k的点的最小代价。
- 转移：
- (i)枚举分界点，合并区间
- (ii)在x轴区间$[i,j]$放一个尽量大的矩形,若能覆盖区间中所有高度<k的位置，则可以转移到$F[i][j][1]$
- $O(n^4)$

Tourism

给定一个 n 个点，m 条边的无向图，在第 i 个点建立旅游站点的费用为 C_i。在这张图中，任意两点间不存在节点数超过 10 的简单路径。

请找到一种费用最小的建立旅游站点的方案，使得每个点要么建立了旅游站点，要么与它有边直接相连的点里至少有一个点建立了旅游站点。

- $2 \le n \le 20000, 0 \le m \le 25000$。
- Source：POI 2014

求最小代价

Hint

- 随便建一棵dfs树的深度不超过10
- 并且因为是无向图所有所有边都是返祖边。

sol

- 一般都会把子树划分状态，但这题我们换个想法。
- 设$F[i][S]$表示确定完**dfs序**从1..i的点，i到根的覆盖情况是S的最小代价。
- 也可以理解为dp状压的是一个状态栈

- 考虑子树内对外的影响， 点i到根上的点有可能已经被覆盖。
- 考虑外面对子树内的影响，i到根上的点 会 覆盖子树内的一些点。
- 因此,S是一个三进制状态，表示从根到x这些点，选了/没选已被覆盖/没选没被覆盖。

- S只需要开到深度位，可以通过随机取根的方式来避免被卡

- 转移就是维护状压的栈，然后保证出栈的点都已经满足要求即可。
- 时间上界$(n+m)*3^{10}$，但是远达不到。

JZOJ4019 path

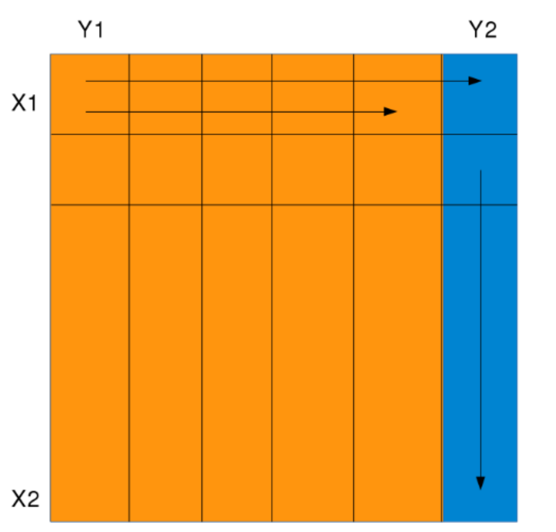
- 给定一个 n* m 的网格，你在左下角 (n,1)，一开始你面向上方，你只能往前走或者右拐，障碍和走过的点不能走。给出x,y, 求走到 (x,y) 的方案数 mod k 的值。

- 也就是能一直向右转圈。

- N,m<=100
- k<=10^9

sol

- 反过来想，从终点出发，矩形在一圈圈扩大。
- 设状态f[a][b][x][y][0..3]表示从矩形(a,b,x,y)的左上角，右下角，左下角，右下角进入这个矩形，往后一直在这个矩形内运动，到达终点的方案数。
- 转移？枚举当前方向走了多少步再拐弯
- $O(n^4 \cdot n)$，不太行

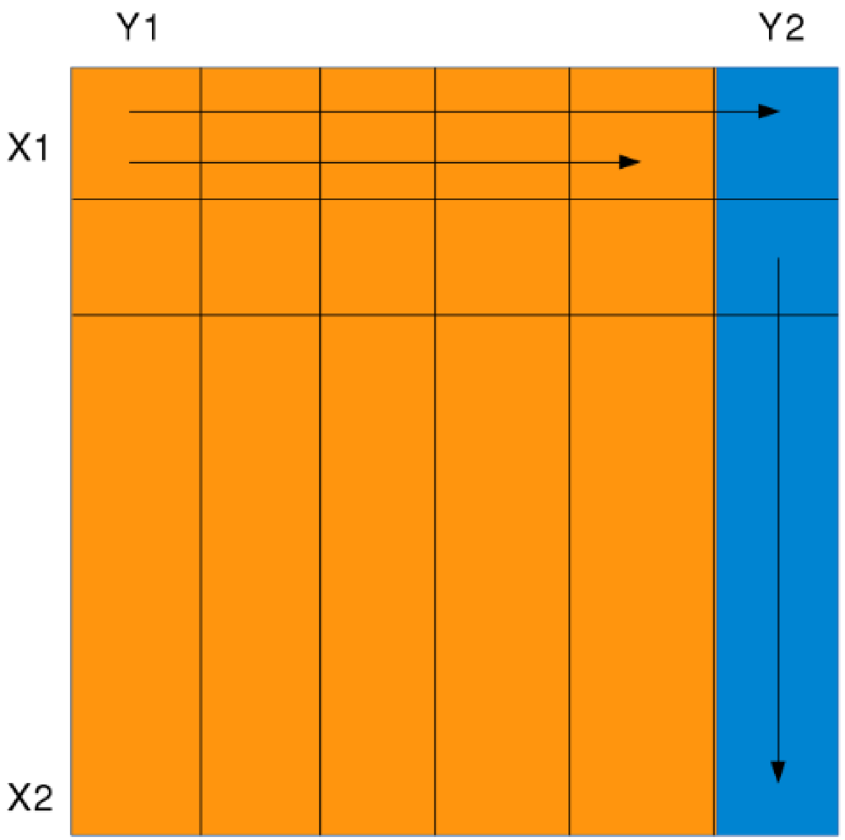


Transfers of f[0][x1][y1][x2][y2-1] and f[0][x1][y1][x2][y2] differs exactly 1 row, which is f[1][x1+1][y1][x2][y2]

Improve it

- 找一下子问题
- 比方说现在转移整个矩形右上角的情况。
- 只需要在黄色的子矩形基础上多处理从蓝色那边走的情况。
- 这实际上是去掉当前这行的矩形

- 可以O(1)转移了



Transfers of f[0][x1][y1][x2][y2-1] and f[0][x1][y1][x2][y2] differs exactly 1 row, which is f[1][x1+1][y1][x2][y2]

转移的优化

- 注意题目的性质
- 数据结构
- 决策单调性
- 单调队列
- 斜率优化
- 凸优化
- 长链剖分
- 四边形不等式
- …

Paint Pearls

给定一个长度为 n 的序列，每一位有一个目标颜色。初始时每一位都没有颜色。每次可以选择一个区间，将区间内的所有元素改为其**目标颜色**。设区间内不同颜色的数量为 x ，则操作的代价为 x^2 。求最小代价。

$n \le 50000$ 。 □

Paint Pearls - Solution

容易发现答案的上界为 n —每次操作一个元素即可。

基于上界又可以发现，如果一个区间内有超过 $\sqrt{n}$ 种颜色，我们一定不会去操作它。

换句话说，对于一个 $f[i]$ ，只需要考虑 $\sqrt{n}$ 个不同的 $w(j,i)$ 的取值。

而 f 显然单调不减，因此 $w(j,i)$ 相同时选择最靠左的 $f[j-1]$ 。

我们只需知道这些 j 的值。 □

只要记录当前位置往左出现的前 $\sqrt{n}$ 种颜色，及对应位置即可。

移动到下一个数时，如果颜色出现在了前 $\sqrt{n}$ 种之中，则暴力删除并移到最前面。

总复杂度 $O(n\sqrt{n})$ 。

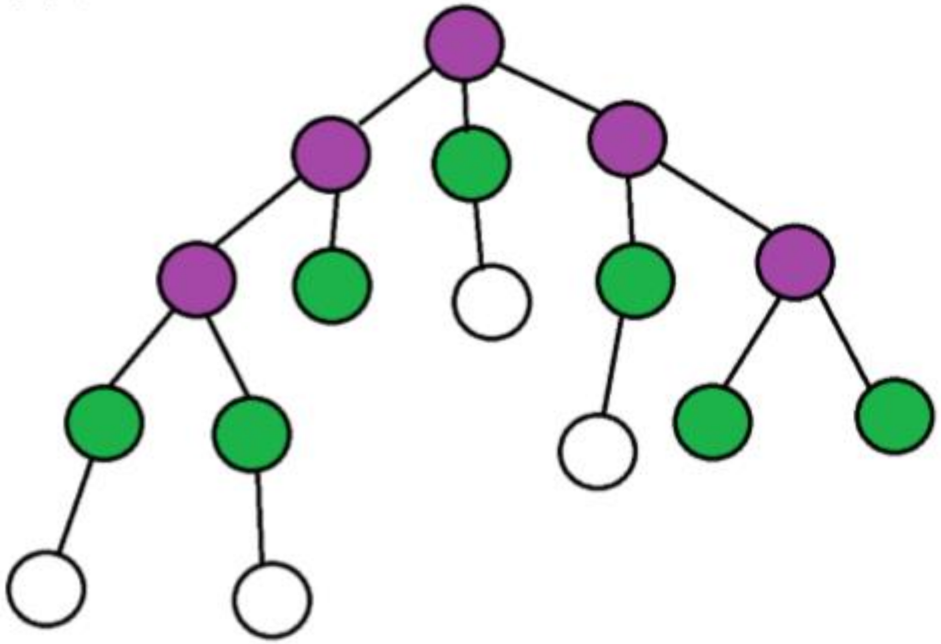
Tree chain problem

给定一棵有 n 个点的树，以及 m 条树链，其中第 i 条树链的价值为 w_i，请选择一些没有公共点的树链，使得价值和最大。

- $1 \leq n, m \leq 100000$。
- Source：2015 Multi-University Training Contest 1

Tree chain problem

- 考虑树形 DP，设 $f(x)$ 为以 x 为根的子树内选取不相交树链的价值和的最大值，枚举一条 LCA 为 x 的链 u, v, w，那么当前方案的价值为 $w+$ 去除 u 到 v 路径上的点后深度最小的点的 f 的和。
- 如图，紫色部分为 u 到 v 路径上的点，绿色部分计入当前贡献：



- 建dfs序树状数组，维护单点修改与可差分的树链查询。

Data Structure You've Never Heard Of

给定一个长度为 n 的序列 $a_1, a_2, \ldots, a_n$，每个元素都是一个 d 维 01 向量，求所有不下降子序列的个数。

对于 d 维向量，$a_i \le a_j$ 等价于 a_i 的每一维都不大于 a_j。

- $1 \le n \le 200000, 1 \le d \le 16$。
- Source：ftiasch's Contest #4

sol

- $a \le b$等价于$a|b = b$
- 设$f[i]$表示以i结尾的个数，直接dp，O(n^2)
- 考虑维护高维前缀和，均摊一下修改与查询的复杂度。
- Sum[a][b]表示前8位是a，后8位是b的子集的位置的f和。
- 查询：枚举前8位，$O(2^{\mathrm{d}/2})$
- 修改：…
- $O\left(n2^{\frac{d}{2}}\right)$


氷

斜率优化

- NOI2019day1T1 ~~(加强~~
- 有n个点，m辆火车
- 每辆火车有给定的起点、终点，出发、到达时间
- 现在是零时刻，你要从1号点到n号点
- 每一段**不**坐火车的时间t会带来At^2+Bt+C的代价。
- 求出每一段不坐火车的时间代价之和 + 最后到达时间的最小值。
- N<=1e5,m<=2e5
- 1s

设好状态就行了

- 你会发现设f[i]表示坐完第i辆火车后的最小时间特别好dp
- M^2的dp很简单，70pts

维护直线

- 观察一下转移方程（假如是i转移到j），发现有一个项与i，j都相关。
- 而且是直线kx+b的形式。对于不同的j只是其中的x不同。

- 在每个点上维护直线上凸壳就可以了。
- 直线插入的斜率是单调的
- 用单调栈就行，查询就二分最优直线。
- O(m log m)

- 似乎查询的横坐标也是递增的，用单调队列维护凸壳就可以O(m)了

经典题

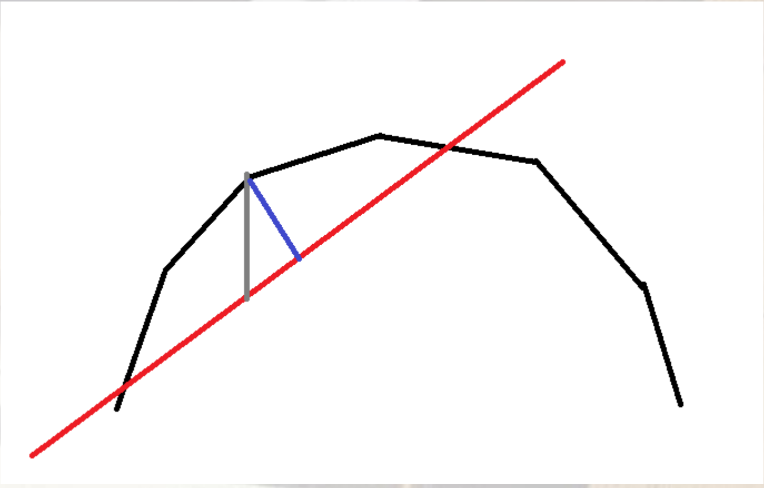
- 一个长度为n的序列，求出选**恰好**k个不相交区间的最大权值和。
- K,N<=1e5
- -1e9<=ai<=1e9

- 设函数f(z)表示取z段的最大值。
- 观察发现，F(z)上凸。证略。

凸优化

- 基于两个事实：
- 1. 若取一段**代价**是c，任意段数地取，取到最大收益W时是取z段。w+cz就是恰好取z段的最大值。

- 2. 考虑求f(z)-cz（黑色-红色）的最大值，就相当于引一条斜率为c的切线从无穷远处切下来所切到的点。



凸优化

- 由于函数是凸的，每个点都会被切到。
- 通过二分调整代价c，使他恰好切到询问的k.
- 求到的最大值去除取每段的代价就是答案。

一些常见模型

- 排列dp
- 数位dp
- ~~插头dp~~
- 动态dp

- 仙人掌/圆方树上dp
- 树型dp，DAG上dp


氷

Shopping

给定一棵有 n 个点的树，第 i 个点有 d_i 件商品，价格为 c_i，价值为 w_i。

你手头有 m 块钱，且你要保证你买过的点在树上互相连通，问买到的物品的总价值最多是多少。

- $1 \leq n \leq 500, 1 \leq m \leq 4000, d_i \leq 100$。
- Source: BZOJ 4182

假如定一个根，就是树形依赖多重背包

- 树形依赖01背包：在dfs序上做，O(nv)，跟刚才那题类似。

【分析】

这题有多种做法，如果直接背包合并，复杂度是$O(nm^2)$，难以承受。

其中一种改进方法是在DFS序上做，那么每次要么跳过一段子树，要么继续往下，复杂度$O(nm)$，这也是经典trick。

还有一个trick是zzy7在校内训练提出的方法，跟要讲的这题有异曲同工之妙，在这里感谢zzy7（虽然可能已经AFO了）

我们考虑正常背包合并，复杂度之所以为$O(nm^2)$，因为每次要合并背包。

是否可以在做的时候，把父亲的dp值传递给儿子，然后由儿子继续父亲的dp，然后做下去呢？答案是可以的。

我们修改表示方式为做完了上面那条链和下面的子树的答案，每次做两件事情：把父亲的传给儿子做，把儿子做完的传上来。

复杂度同样是$O(nm)$。

Shopping

- 考虑某个点必选的情况，只要以这个点为根进行树形依赖背包即可，具体做法不再赘述，通过二进制拆分可以做到 $O(nm\log d)$。
- 对这棵树进行点分治，重心要么选，要么不选，第一种情况用上述方法 DP 即可，第二种情况则是子问题，递归处理即可。
- 时间复杂度 $O(nm\log n\log d)$。

GDSOI2019 D2 T1 高中生数学题

小明知道老师喜欢将简单的题目改点条件变成第二天的作业题，于是开始提前开始思考对于任意数字 m，杨辉三角第 n 行有多少 m 的倍数。这个问题超出了小明的能力，小明将其简化为，m 只包含一个质因子，也就是可以写成 $m = p^k$ 的形式，其中 p 是一个质数。

形式化地描述这个问题，即给定 n, m，求 $C_n^i, 0 \le i \le n$ 中有多少个数是 m 的倍数。

Data Constraint

对于前 15% 的数据，n ≤ 5000
对于前 35% 的数据，n ≤ 2 · 10^7
另外有 20% 的数据，k = 1
对于 100% 的数据，k ≥ 1,1 < p^k ≤ n ≤ 10^18 .

sol

- 就是要含有p的幂次>=k

- 库默尔定理:
- C_n^i含有p的幂次
- =i+(n-i)在p进制下的进位次数。

- 证明：C(n,i)=n!/i!/(n-i)!，所以其中p的幂次
- $= \sum \left(\frac{n}{p^k}\right) - \left(\frac{i}{p^k}\right) - \left(\frac{n-i}{p^k}\right)$，全部下取整

- 也就是n的p进制去掉后k位减去i的再减去n-i的。
- 若n的第k位没有被i+(n-i)的低位进位，则=0.否则=1。
- 因此是i+(n-i)的进位次数。

- 于是问题变成了，存在多少i满足i+(n-i)进位k次以上。
- 也就是n-i借位k次以上。

- 从高位往低位做数位dp
- 设状态
- F[第i位][前面是否完全与n相同][借位次数][是否被下一位借位]
- 枚举当前位填什么，转移即可。
- 当然不能真的枚举。最后总数减掉借位k次以内的。
- O(logn^2)

NOI 7.12 抽鬼牌

- 求1~n的不存在长度大于等于m的公差=1或-1的连续等差数列的排列数。
- N,m<=200
- 1s, 512mb

笛卡尔树与排列

- 据说有挺多做法
- 想到其他做法的大佬可以来口胡一下。

- 考虑从小到大加入所有数，那么可以发现你只关心上一个数所在的等差数列的长度。
- **每个排列的带标号笛卡尔树是唯一的，于是我们可以计算合法的笛卡尔树数量**
- 设f[插入了1~i][i所在等差数列长度][笛卡尔树数][0/1/2/3]
- 其中0/1/2/3分别表示i在其笛卡尔树的中间，左边，右边，或者单独一块。(假如一颗笛卡尔树看成他的中序遍历)

转移

- 插入i的时候就是在构造以i为根的笛卡尔树
- 讨论一下儿子的情况就可以了
- 注意控制等差数列长度<=m


氷

NOI 7.13 MIS

- 一棵树，每个点有正或负的权值c。
- 一个排列的价值是按顺序删点，并且将当前点上的权值**加给**与其相连的其他点。将每个点被删时的权值求和即为价值。
- 请求出排列的最大价值。
- N<=400，c<=1e9

SOL

- 涉及到操作树上点的顺序的时候
- 可以考虑一下边的定向
- 这里把先操作的点连向后操作的。
- 一波操作之后，就可以发现本质是给树边定向，然后一个点的权值就是**能到达他的所有点**的权值和。（任意一种定向方案都有对应的一些排列），求最大定向方案价值。

- 这个东西挺好dp的吧，有没有大佬上来港一下

- 设f[i][j][k]表示以i为根的子树，i往下能到j个点，能到i的部分的权值被多计算了k倍。
- 讨论每条边的方向，暴力转移就可以了。答案是f[1][j][0]
- 由于第二维是O(size)的，总体复杂度是O(n^3)。

- 完

JOI 2019春季合宿 cake3

- 你有两个数组w[n]和loc[n]。保证loc递增，一个区间[L,R]的价值是区间内前m大的w之和 $-\ loc[R]-loc[L]$
- 求所有长度大于等于m的区间的最大价值。

$3 \leq \mathrm{M} \leq \mathrm{N} \leq 200\,000$

$1 \leq \mathrm{Y_i} \leq \mathrm{C_i} \leq 1000\,000\,000$

测试点编号	N ≤	分值
1	100	5 $points$
2	2000	19 $points$
3	200000	76 $points$

决策单调性

- 可以发现，对于任意右端点R的最优左端点是不下降的。
- 大致证明:
- 设g(x)是右端点x对应的最优左端点，假如$g(y)<g(x)<x<y$
- 设dis(l,r)是一个区间跨过的距离，f(l,r)是这个区间里前m大之和。
- 那么意味着区间[g(y),y]优于[g(x),y]
- $dis(g(y),y)+f(g(y),y)\ge dis(g(x),y)+f(g(x),y)$，于是有
- $dis(g(y),x)+f(g(y),x)\ge dis(g(x),x)+f(g(x),x)$
- (是因为f有如下性质：若$f(A)\ge f(A')$，A'是A的子集，那么$f(A-x)\ge f(A'-x)$.)
- 于是[g(y),x]优于[g(x),x]，假设不成立。

- 考虑一个分治solve([x,y],[L,R])
- 表示现在要求给定左端点取值区间与右端点取值区间的这一段答案。
- 每次取右端点的mid，配合主席树找到[x,y]中mid对应的最优决策。
- 这里是O((y-x)*log n)
- 对[L,mid-1],[mid+1,R]分治下去即可。每一层分治中[x,y]的和都是O(n)
- O(n log^2 n)

长链剖分

- 重链剖分：选重儿子
- 长链剖分：选深度最大的儿子

- 应用:
- O(nlog)预处理，O(1)询问第k级祖先
- 快速合并深度为下标的子树信息

牛客OI周赛11C

- 一棵n个点的树。
- 你可以选择深度为d以内的任意L个点。
- 最大化这些点到根的链并大小。要对d=1..n做出回答。

20% $n \le 300$，$m \le 100$

40% $n \le 3000$，$m \le 1000$

80% $n \le 3 * 10^5$，$m \le 10^5$

100% $n \le 3 * 10^6$，$m \le 10^6$

对于所有数据 $1 \le l \le n$，$0 \le d \le n-1$

N方

- 也就是不考虑d的限制的O(n)做法。
- 设dep[x]表示x往下最深链的长度。
- cnt[x]为dep=x的点数。
- 答案为$\sum \min(cnt[x], l)$

- 正确性:
- 设f(0)表示原树的答案，f(1)表示剥去一层叶子的树答案
- 则f(k)=f(k+1)+min(cnt[k+1],l)

每个d都做太慢了

- 我们分深度加入所有点。
- 可以发现每加一层点，他们所形成的虚树的dep会+1。
- 每个点向上一个个点更新。
- 不更新的点都已经被这一层的其他点更新过了。因此加入点x时要更新的事实上是x~lca(x,last).
- 这是一段dep连续的链。因此cnt只有两个地方要改动。
- 可以O(1)求lca
- O(n log)或者O(n alpha)
- 但是tarjan求lca好长啊，~~还爆栈啊~~

长链剖分

- 在某个点的父亲考虑他的贡献
- 可以发现，加入一层点(深度为T)的时候。假如点p有k个子树内有这层的点，那他对cnt$[T-T_p]$有k-1的贡献。

- 换句话说，**去除掉长儿子之后**，每有一个子树有深度为k的点就对cnt$[T-T_p]$有1的贡献。
- 暴力打标记做就可以了。
- 遍历每个点的轻儿子所在重链是O(n)的。