数学场

四题分别考了纯数论、利用概率期望的性质做DP、容斥计数、组合数学计数DP四个知识点，整体难度在NOIP到省选间。

热身两题为容斥和推式子。

T0 array

原题：bzoj 数列的gcd

原题要求gcd恰好为d的方案数，这不太好求；我们的想法是类似莫比乌斯反演，求==d转换成$\Sigma_{d|p}ans[d]$，记为$f[p]$；然后莫比乌斯反演$ans[d]=\sum_{d|p}f[p]\mu[p/d]$，计算所有答案是一个调和函数。

再考虑$f[p]$如何求，我们还有一个恰好K个不同，n-K个相同的性质，首先预处理出来$num[p]$表示是p的倍数的数的个数，这k个不同的数一定是在num[p]里取到的，即答案最后长这样：

$$f(p)=\binom{num_p}{n-k}(\lfloor\frac{m}{p}\rfloor-1)^{num_p-(n-k)}\lfloor\frac{m}{p}\rfloor^{n-num_p} \tag{1}$$

意思是首先在num_p个数里选出相同的几个，则剩下如果a_i是num_p里的，要满足和a_i不同且是p的倍数；如果a_i不是num_p里的，就是直接任意填就好了。

也可以不莫比乌斯反演，直接做的时候$ans(p)=f(p)-\sum_{d|p,d<p}ans(d)$即可。

总复杂度$O(nlogn)$

T0.5 ring

from hdu多校。

一道容斥+推式子的题。

首先考虑答案是怎么来的：由于n的顺序影响最终的答案，而m的顺序不影响，我们可以先求出任意一个n的排列，然后把它分成m段，保证每段长度不超过k的方案数，记录为$f[n][m][k]$；而这样的话，将m的顺序也考虑进答案了，我们除一个 m！系数即可。

所以最后答案 = $f[n][m][k]*n!\ /m!$

考虑$f[n][m][k]$怎么求：如果没有k的限制，相当于在n-1个缝隙里面选m-1个作为拆分的断点，答案是$\binom{n-1}{m-1}$；加上每一个都$\leq k$的限制的话，一个自然的想法是考虑容斥。假设至少有i个不满足限制为g[i]。恰好有i个不满足限制为f[i]的方案数。

我们有$g[i]=\sum_j\binom{j}{i}f[j]$

二项式反演有$f[i]=\sum_j(-1)^{j-i}\binom{j}{i}g[j]$

而g[i] = $\binom{n-1-ki}{m-1}\binom{m}{i}$。意思是枚举顺序排列里哪几个不符合限制；其他点不管是否符合性质，一种枚举下对答案的贡献记为第一个组合数。

最终f[0]就是要求的东西。

总复杂度$O(nlogn)$

T1 math

from hdu，比较常规的一道筛题。

结论：$(2^i-1,2^j-1)=2^{(i,j)}-1$。

证明：$(2^i-1,2^j-1)=(2^i-1,2^j-2^i)=(2^i-1,2^i(2^{j-i}-1))=(2^i-1,2^{j-i}-1)$。

归纳即可。

令$f(x)=(2^x-1)^k$.

原式可以化简为：

$$\begin{aligned}\Sigma_{i=1}^{n}\Sigma_{j=1}^{n}f[(i,j)] &= \Sigma_{d=1}^{n}\Sigma_{i=1}^{n}\Sigma_{j=1}^{n}f[d][(i,j)==d]\\ &= \Sigma_{d=1}^{n}f(d)\Sigma_{i=1}^{[n/d]}\Sigma_{j=1}^{[n/d]}\Sigma_{t|(i,j)}\mu(t)\\ &= \Sigma_{d=1}^{n}\Sigma_{t=1}^{[n/d]}(\lfloor\frac{n}{dt}\rfloor)^2\mu(t)\\ &= \Sigma_{T=1}^{n}(\lfloor\frac{n}{T}\rfloor)^2\Sigma_{d|T}f(d)\mu(\frac{T}{d})\end{aligned}$$

令$g=f*\mu$，则$g*1=f*\mu*1=f*\epsilon=f$。由于g是积性函数，f的前缀好求，1的前缀好求，则g可用杜教筛。

$\Sigma_{i=1}^{n}f(i)=\Sigma_{i=1}^{n}\Sigma_{k=0}^{K}\binom{n}{k}2^{ik}(-1)^{K-k}=\Sigma_{k=1}^{K}\binom{K}{k}(-1)^{K-k}\Sigma_{i=1}^{n}2^{ik}$ 等比求和即可。

设g的前缀和为$S(n)$。

有$S(n)=\Sigma f(i)-\Sigma_{i=2}^{n}S(\lfloor\frac{n}{i}\rfloor)$

$g(n)$的前缀和可以算之后，原式对T做整除分块即可。

T2 trip

CF605E

我们想求出E[i]表示从i号点出发到达n的期望时间。

根据贪心的思想，我们可以写出转移的式子为：

$$E[i]=1+\sum_{Ej<Ei}Ej*pi,j*\prod_{Ek<Ej}(1-pi,k)+E_i*\prod_{Ej<Ei}(1-pi,j) \tag{2}$$

意思是尽量走到期望小的点。

然后可以暴力高斯消元解方程。

也可以把它化简：

$$E[i]=\frac{1+\sum_{Ej<Ei}Ej*pi,j*\prod_{Ek<Ej}(1-pi,k)}{1-\prod_{Ej<Ei}(1-pi,j)} \tag{3}$$

我们记下面$\prod_{Ej<Ei}(1-pi,j)$为gi，上面的$\sum_{Ej<Ei}Ej*pi,j*\prod_{Ek<Ej}(1-pi,k)$为$fi$。

考虑每一个点的f和g都会被比他小的点更新，那么我们的想法就是按期望从小到大依次加入，这样就消除了后效性，确保每时每刻拿来更新的$E[i]$都是合法的。

具体而言，一开始$E[n]=0$，我们类似dijkstra的想法，每次选择一个最小的点出来作为j。

第一步：更新其他i的$f[i]$，由于此时只添加了所有$E_k<E_j$的k，所以此时$f[i]$的增量恰好可以用$E_j*p(i,j)*g[i]$来表示。

第二步：更新其他i的$g[i]$，然后再更新$E[i]$。

总复杂度$O(n^2)$。

T3 cure

51nod zbox的刷题2

考虑分别计算$1,2,3,...,n$的每个非空子集s对最终答案的贡献。经过i轮后s内每个元素仍全部存活的概率是$p^{i|s|}$，假设这个集合贡献答案的系数为$b|s|$（显然这个系数是只与集合大小有关的），

那么最终答案就是：$\sum_{i=1}^{n}\binom{n}{i}bi\sum_{j=0}^{\infty}p^{ji}$

原式$=\sum_{i=1}^{n}\binom{n}{i}bi\frac{1}{1-p^i}$

考虑如何计算bi，只需要考虑贡献是如何计算的即可。

一个大小为i的集合一次应当被计算的贡献为ai，实际上为$ai=\sum_{j=1}^{i}\binom{i}{j}bj$。

即$bi=\sum_{j=1}^{i}(-1)^{j-i}\binom{i}{j}aj$，二项式反演即可求出$b$。

复杂度$O(nlogn)$.

T4

loj 博弈论和概率统计

组合 + 莫队

首先这个p是假的，没有任何作用，因为我们最后是求方案数。

我们考虑它的得分的组合意义，如果没有限制，那么最终得分一定是$N-M$；否则，得分为$N-M+T$。其中T为分数等于0的时候输的次数。

那么我们把输看成+1，赢看成-1，即求最大前缀和；如果把它投影到网格行走上，就是从(0,0)走到(n,m)，且T为y-x的最大值。

对于某一个特定的最大值，我们不太好求；但是我们可以求出$T\geq i$的值，对应的是经过y=x+i直线的方案数。

而经过y = x+i的方案数，为$\binom{N+M}{M-i}$*，记录为H[i]。

所以对于一组询问，答案需要求解一个类似这样的东西：

$\sum i[T==i]=\sum[T>=i]=\sum\binom{N+M}{M-i}$

即我们需要快速求解$f(x,y)=\Sigma_{i=0}^{x}\binom{y}{i}$

把x,y看成两维坐标，我们用莫队的想法，需要快速处理$f(x,y)->f(x,y\pm1)$或者$f(x,y)->f(x\pm1,y)$

后者好求，相当于多一个组合数；后者相当于把前缀往下平移一格；我们利用$\binom{y}{x}=\binom{y-1}{x}+\binom{y-1}{x-1}$，即我们求出f(x-1,y)，f(x-1,y)+f(x,y)=f(x,y+1)，也可以O(1)求。

所以总复杂度$O(n\sqrt{n})$。其中n为值域。

*补充：一类卡特兰数的求法

最简单的问题：从（0,0）出发，往右上走，使得不跨越过y=x 路径到达(m,n)的方案数。

思路：像这种问题我们的做法一般是容斥。

我们做(0,0)关于y=x的对称点，为(-1,1)。

我们发现任意一条从起点到终点，且跨越线的的路径可以和一条对称点到终点的路径一一对应。

(-1,1)随便走的答案为(n+m,n-1)。则总答案为(n+m,n)-(n+m,n-1)

用这种方法也可以处理处不跨过y=x+i的方案数。