数学相关

Cai

cssyz

2017 年 3 月 1 日

Contents

一般形式

一般形式

- $f_n = \sum_{i=1}^{k} a_i * f_{n-k}$

一般形式

- $f_n = \sum_{i=1}^{k} a_i * f_{n-k}$
- 例子：fibonacci数列：$f_n = f_{n-1} + f_{n-2}$

特征方程

特征方程

- 这里只说明形如$f_n = pf_{n-1} + qf_{n-2}$的形式的

特征方程

- 这里只说明形如$f_n = pf_{n-1} + qf_{n-2}$的形式的
- 特征方程为$x^2 = px + q$

特征方程

- 这里只说明形如$f_n = pf_{n-1} + qf_{n-2}$的形式的
- 特征方程为$x^2 = px + q$
- 设特征方程可解出两个根s, t

特征方程

- 这里只说明形如$f_n = pf_{n-1} + qf_{n-2}$的形式的
- 特征方程为$x^2 = px + q$
- 设特征方程可解出两个根s, t
- 则数列通项公式为$f_n = as^n + bt^n$，a, b的值由数列的任意两项决定

特征方程

- 这里只说明形如$f_n = pf_{n-1} + qf_{n-2}$的形式的
- 特征方程为$x^2 = px + q$
- 设特征方程可解出两个根s, t
- 则数列通项公式为$f_n = as^n + bt^n$，a, b的值由数列的任意两项决定

特征方程

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$
- 将上式还原成一个递推式，设其通项公式为$f_n = xs^n + yt^n$

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$
- 将上式还原成一个递推式，设其通项公式为$f_n = xs^n + yt^n$
- 则$x = 1, s = \frac{b+\sqrt(d)}{2}, t = \frac{b-\sqrt(d)}{2}$

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$
- 将上式还原成一个递推式，设其通项公式为$f_n = xs^n + yt^n$
- 则$x = 1, s = \frac{b+\sqrt(d)}{2}, t = \frac{b-\sqrt(d)}{2}$
- 进一步$f_0 = 1 + b, f_1 = \frac{(1+y)b}{2} + \frac{(1-y)\sqrt{d}}{2}$，显然$y$取1好算

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$
- 将上式还原成一个递推式，设其通项公式为$f_n = xs^n + yt^n$
- 则$x = 1, s = \frac{b+\sqrt(d)}{2}, t = \frac{b-\sqrt(d)}{2}$
- 进一步$f_0 = 1 + b, f_1 = \frac{(1+y)b}{2} + \frac{(1-y)\sqrt{d}}{2}$，显然$y$取1好算
- 由s, t是方程$x^2 - px - q = 0$的两根，于是$p = b, q = \frac{d-b^2}{4}$

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$
- 将上式还原成一个递推式，设其通项公式为$f_n = xs^n + yt^n$
- 则$x = 1, s = \frac{b+\sqrt(d)}{2}, t = \frac{b-\sqrt(d)}{2}$
- 进一步$f_0 = 1 + b, f_1 = \frac{(1+y)b}{2} + \frac{(1-y)\sqrt{d}}{2}$，显然$y$取1好算
- 由s, t是方程$x^2 - px - q = 0$的两根，于是$p = b, q = \frac{d-b^2}{4}$
- 矩阵乘法算出f_n，$s^n = f_n - t^n$，其中$-1 < t \le 0$

特征方程

- 例题：求$\lfloor(\frac{b+\sqrt(d)}{2})^n\rfloor(mod\ 7528443412579576937)$
- 将上式还原成一个递推式，设其通项公式为$f_n = xs^n + yt^n$
- 则$x = 1, s = \frac{b+\sqrt(d)}{2}, t = \frac{b-\sqrt(d)}{2}$
- 进一步$f_0 = 1 + b, f_1 = \frac{(1+y)b}{2} + \frac{(1-y)\sqrt{d}}{2}$，显然$y$取1好算
- 由s, t是方程$x^2 - px - q = 0$的两根，于是$p = b, q = \frac{d-b^2}{4}$
- 矩阵乘法算出f_n，$s^n = f_n - t^n$，其中$-1 < t \leq 0$
- 如果$d \neq b^2$ and $n \equiv 0(mod\ 2)$，则$ans = f_n - 1$，否则$ans = f_n$

计算公式

计算公式

- $(A * B)[i][j] = \sum_{k=1}^{n} A[i][k] * B[k][j]$

计算公式

- $(A * B)[i][j] = \sum_{k=1}^{n} A[i][k] * B[k][j]$
- 时间复杂度$O(n^3)$

优化线性递推

优化线性递推

- 线性递推:形如$f_n = \sum_{i=1}^{k} a_i * f_{n-i}$

优化线性递推

- 线性递推:形如$f_n = \sum_{i=1}^{k} a_i * f_{n-i}$
- 时间复杂度$O(k^3 logn)$

优化线性递推

- 线性递推:形如$f_n = \sum_{i=1}^{k} a_i * f_{n-i}$
- 时间复杂度$O(k^3 logn)$
- 当k较小但n很大的时候使用

优化动态规划

优化动态规划

- 形如$f[i][j]=\sum_{k=1}^{n}f[i][k]*g[k][j]$的动态规划

优化动态规划

- 形如$f[i][j] = \sum\limits_{k=1}^{n} f[i][k] * g[k][j]$的动态规划
- 时间复杂度$O(n^3 logm)$

优化动态规划

- 形如$f[i][j]=\sum_{k=1}^{n} f[i][k]*g[k][j]$的动态规划
- 时间复杂度$O(n^3 logm)$
- 例题：HNOI　GT考试

矩阵乘法与图论

矩阵乘法与图论

- floyd算法判断两点是否联通$L[i][j] = \sum_{k=1}^{n} L[i][k] \&\& L[k][j]$

矩阵乘法与图论

- floyd算法判断两点是否联通$L[i][j] = \sum_{k=1}^{n} L[i][k] \&\& L[k][j]$
- 时间复杂度$O(n^3)$

矩阵乘法与图论

- floyd算法判断两点是否联通$L[i][j] = \sum_{k=1}^{n} L[i][k] \&\& L[k][j]$
- 时间复杂度$O(n^3)$
- 判断两点是否可以经过k条边到达：矩阵乘法

矩阵乘法与图论

- floyd算法判断两点是否联通$L[i][j] = \sum_{k=1}^{n} L[i][k] \&\& L[k][j]$
- 时间复杂度$O(n^3)$
- 判断两点是否可以经过k条边到达：矩阵乘法
- 时间复杂度$O(n^3 logk)$

矩阵乘法与图论

- floyd算法判断两点是否联通$L[i][j] = \sum_{k=1}^{n} L[i][k] \&\& L[k][j]$
- 时间复杂度$O(n^3)$
- 判断两点是否可以经过k条边到达：矩阵乘法
- 时间复杂度$O(n^3 logk)$

矩阵乘法与图论

矩阵乘法与图论

- 例题: 给出一张$n \le 20, m \le 60$的图，问你在图上走t条边(相邻两次走过的边不能相同)之后从A到B的方案数

矩阵乘法与图论

- 例题: 给出一张$n \leq 20, m \leq 60$的图，问你在图上走t条边(相邻两次走过的边不能相同)之后从A到B的方案数
- 由于相邻两次走过的边不能相同，所以要转化边与点的关系

矩阵乘法与图论

- 例题: 给出一张$n \le 20, m \le 60$的图，问你在图上走t条边(相邻两次走过的边不能相同)之后从A到B的方案数
- 由于相邻两次走过的边不能相同，所以要转化边与点的关系
- 即边化为点，点化为边

矩阵乘法与图论

- 例题: 给出一张$n \leq 20, m \leq 60$的图，问你在图上走t条边(相邻两次走过的边不能相同)之后从A到B的方案数
- 由于相邻两次走过的边不能相同，所以要转化边与点的关系
- 即边化为点，点化为边
- 矩阵乘法即可

其他

其他

- 见我博客

其他

- 见我博客
- http://blog.csdn.net/cuso45h20/article/details/53439641

素数

- 线性筛 $O(n)$

素数

- 线性筛 $O(n)$

欧拉函数

欧拉函数

- $x=\prod_{i=1}^{t}p_i^{a_i}$，其中$p_i$是质数,则$\phi(x)=x\prod_{i=1}^{t}(1-\frac{1}{p_i})$

欧拉函数

- $x=\prod_{i=1}^{t} p_i^{a_i}$，其中$p_i$是质数,则$\phi(x)=x\prod_{i=1}^{t}(1-\frac{1}{p_i})$
- 积性函数:如果$gcd(a,b)=1$，则$\phi(a*b)=\phi(a)*\phi(b)$

欧拉函数

- $x = \prod_{i=1}^{t} p_i^{a_i}$，其中$p_i$是质数,则$\phi(x) = x\prod_{i=1}^{t}(1-\frac{1}{p_i})$
- 积性函数:如果$gcd(a,b)=1$，则$\phi(a*b)=\phi(a)*\phi(b)$
- 求$\phi(i), i \in [1,p]$：线性筛$O(n)$

欧拉函数

- $x=\prod_{i=1}^{t} p_i^{a_i}$，其中$p_i$是质数,则$\phi(x)=x\prod_{i=1}^{t}(1-\frac{1}{p_i})$
- 积性函数:如果$gcd(a,b)=1$，则$\phi(a*b)=\phi(a)*\phi(b)$
- 求$\phi(i), i\in[1,p]$：线性筛$O(n)$
- 欧拉定理，如果$gcd(a,p)=1$，则$a^{\phi(p)}\equiv 1(mod\ p)$

最大公约数

最大公约数

- 欧几里得算法

最大公约数

- 欧几里得算法
- $O(logn)$

最大公约数

- 欧几里得算法
- $O(logn)$
- 更相减损术

最大公约数

- 欧几里得算法
- $O(logn)$
- 更相减损术
- 例题Codevs2305 SuperGcd

乘法逆元

乘法逆元

- 拓展欧几里得算法 $O(logn)$

乘法逆元

- 拓展欧几里得算法　$O(logn)$
- 欧拉定理：如果$gcd(a, b) = 1$，则$a^{\phi(b)} \equiv 1(mod\ b)$

乘法逆元

- 拓展欧几里得算法 $O(logn)$
- 欧拉定理：如果$gcd(a,b)=1$，则$a^{\phi(b)}\equiv 1(mod\ \ b)$
- 线性求$[1,p]$的逆元

乘法逆元

- 拓展欧几里得算法　$O(logn)$
- 欧拉定理：如果$gcd(a, b) = 1$，则$a^{\phi(b)} \equiv 1(mod\ \ b)$
- 线性求$[1, p]$的逆元
- 线性求$[1, p]$阶乘的逆元: $(x!)^{-1} \equiv (x+1)((x+1)!)^{-1}(mod\ \ p)$

一元一次同余方程

一元一次同余方程

- $ax \equiv b(mod\ c)$

一元一次同余方程

- $ax \equiv b(mod\ \ c)$
- 方程当且仅当$gcd(a, c)|b$时有解

一元一次同余方程

- $ax \equiv b(mod\ \ c)$
- 方程当且仅当$gcd(a,c)|b$时有解
- 解法：将方程转化为$ax+cy=b$的不定方程形式

一元一次同余方程

- $ax \equiv b(mod\ \ c)$
- 方程当且仅当$gcd(a, c)|b$时有解
- 解法：将方程转化为$ax + cy = b$的不定方程形式
- 拓展欧几里得算法求解

BSGS算法

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数
- 利用暴力算法求出$a^0 - a^m(mod\ p)$的值　(baby step)

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数
- 利用暴力算法求出$a^0 - a^m(mod\ p)$的值　(baby step)
- 如果$k \in [m, 2m)$且k是解，则$a^{k-m} \equiv b * (a^m)^{-1}(mod\ p)$

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数
- 利用暴力算法求出$a^0 - a^m(mod\ p)$的值　(baby step)
- 如果$k \in [m, 2m)$且k是解，则$a^{k-m} \equiv b * (a^m)^{-1}(mod\ p)$
- 把$a^0 - a^m(mod\ p)$的值存在哈希表中，每次就可以$O(1)$判断(giant step)

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数
- 利用暴力算法求出$a^0 - a^m(mod\ p)$的值　(baby step)
- 如果$k \in [m, 2m)$且k是解，则$a^{k-m} \equiv b * (a^m)^{-1}(mod\ p)$
- 把$a^0 - a^m(mod\ p)$的值存在哈希表中，每次就可以$O(1)$判断(giant step)
- 显然$m = \lceil\sqrt{p}\rceil$时最优

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数
- 利用暴力算法求出$a^0 - a^m(mod\ p)$的值 (baby step)
- 如果$k \in [m, 2m)$且k是解，则$a^{k-m} \equiv b * (a^m)^{-1}(mod\ p)$
- 把$a^0 - a^m(mod\ p)$的值存在哈希表中，每次就可以$O(1)$判断(giant step)
- 显然$m = \lceil \sqrt{p} \rceil$时最优
- 时间复杂度 $O(\sqrt{p})$

BSGS算法

- 求解$a^x \equiv b(mod\ p)$，其中p是一个质数
- 利用暴力算法求出$a^0 - a^m(mod\ p)$的值 (baby step)
- 如果$k \in [m, 2m)$且k是解，则$a^{k-m} \equiv b * (a^m)^{-1}(mod\ p)$
- 把$a^0 - a^m(mod\ p)$的值存在哈希表中，每次就可以$O(1)$判断(giant step)
- 显然$m = \lceil \sqrt{p} \rceil$时最优
- 时间复杂度 $O(\sqrt{p})$
- 例题：Poj 2417 Discrete Logging(要手打hash表,map有T的风险)

拓展BSGS算法

拓展BSGS算法

- 求解$a^x \equiv b(mod\ c)$，其中c不一定是质数

拓展BSGS算法

- 求解$a^x \equiv b(mod\ c)$，其中c不一定是质数
- $a^x \equiv b(mod\ c)$

拓展BSGS算法

- 求解$a^x \equiv b(mod\ c)$，其中c不一定是质数
- $a^x \equiv b(mod\ c)$
- $p * a^x - q * c = b$

拓展BSGS算法

- 求解$a^x \equiv b(mod\ c)$，其中c不一定是质数
- $a^x \equiv b(mod\ c)$
- $p * a^x - q * c = b$
- 设$d = gcd(a^x, c)$

拓展BSGS算法

- 求解$a^x \equiv b(mod\ c)$，其中c不一定是质数
- $a^x \equiv b(mod\ c)$
- $p * a^x - q * c = b$
- 设$d = gcd(a^x, c)$
- $p * \frac{a^x}{d} - q * \frac{c}{d} = \frac{b}{d}$

拓展BSGS算法

- 求解$a^x \equiv b(mod\ c)$，其中c不一定是质数
- $a^x \equiv b(mod\ c)$
- $p * a^x - q * c = b$
- 设$d = gcd(a^x, c)$
- $p * \frac{a^x}{d} - q * \frac{c}{d} = \frac{b}{d}$
- $\frac{a^x}{d} \equiv \frac{b}{d}(mod\ \frac{c}{d})$

拓展BSGS算法

拓展BSGS算法

- 我们设$x = k + i$，并使得$gcd(a^i, c) = d$

拓展BSGS算法

- 我们设$x = k + i$，并使得$gcd(a^i, c) = d$
- 那么a^k在模$\frac{c}{d}$下是存在逆元的

拓展BSGS算法

- 我们设$x = k + i$，并使得$gcd(a^i, c) = d$
- 那么a^k在模$\frac{c}{d}$下是存在逆元的
- 再套用$BSGS$算法求解即可

拓展BSGS算法

- 我们设$x = k + i$，并使得$gcd(a^i, c) = d$
- 那么a^k在模$\frac{c}{d}$下是存在逆元的
- 再套用$BSGS$算法求解即可
- 例题：Poj2417 Clever Y

孙子定理

- 解线性同余方程组:$x \equiv a_i(mod\ m_i)$，其中m_i两两互质

孙子定理

- 解线性同余方程组:$x \equiv a_i (mod\ m_i)$，其中m_i两两互质
- 令$M = \prod_{i=1}^{n} m_i$，$M_i = \frac{M}{m_i}$，$t_i = M_i^{-1} (mod\ m_i)$

孙子定理

- 解线性同余方程组:$x \equiv a_i (mod\ m_i)$，其中m_i两两互质
- 令$M = \prod_{i=1}^{n} m_i$，$M_i = \frac{M}{m_i}$，$t_i = M_i^{-1} (mod\ m_i)$
- 则方程的解为$x = \sum_{i=1}^{n} a_i t_i M_i$

组合数学

组合数学

- $\binom{n}{m} = \frac{n!}{m!(n-m)!}$

组合数学

- $\binom{n}{m} = \frac{n!}{m!(n-m)!}$
- 组合数递推：$\binom{n}{m} = \binom{n}{m-1} + \binom{n-1}{m-1}$

组合数学与卡特兰数

组合数学与卡特兰数

- $C[n] = \sum_{i=1}^{n-1} C[i] * C[n-i]$

组合数学与卡特兰数

- $C[n] = \sum_{i=1}^{n-1} C[i] * C[n-i]$
- $C[n] = \frac{\binom{2n}{n}}{n+1}$

组合数学与卡特兰数

- $C[n] = \sum\limits_{i=1}^{n-1} C[i] * C[n-i]$
- $C[n] = \frac{\binom{2n}{n}}{n+1}$
- $C[n] = \binom{2n}{n} - \binom{2n}{n-1}$

组合数学与卡特兰数

- $C[n] = \sum_{i=1}^{n-1} C[i] * C[n-i]$
- $C[n] = \frac{\binom{2n}{n}}{n+1}$
- $C[n] = \binom{2n}{n} - \binom{2n}{n-1}$

组合数对质数取模

组合数对质数取模

- 根据$\binom{n}{m} = \frac{n!}{m!(n-m)!}$，用逆元解

组合数对质数取模

- 根据$\binom{n}{m} = \frac{n!}{m!(n-m)!}$，用逆元解
- Lucas定理：$\binom{n}{m} \equiv \binom{n \ mod \ p}{m \ mod \ p}\binom{n/p}{m/p}$

组合数对质数取模

- 根据$\binom{n}{m} = \frac{n!}{m!(n-m)!}$，用逆元解
- Lucas定理：$\binom{n}{m} \equiv \binom{n \ mod \ p}{m \ mod \ p}\binom{n/p}{m/p}$
- 例题:Codevs1830 古代猪文

组合数对任意数取模

组合数对任意数取模

- 线性筛+ 质因数分解+ 快速幂$O(nlogn)$

组合数对任意数取模

- 线性筛+ 质因数分解+ 快速幂$O(nlogn)$
- 例题：HNOI 有趣的数列

组合数对任意数取模

- 线性筛+ 质因数分解+ 快速幂$O(nlogn)$
- 例题：HNOI 有趣的数列

组合数对任意数取模

组合数对任意数取模

- 将模数p质因数分解，将$x \equiv \binom{x}{y} (mod\ p)$打开成$n$个同余方程组：
  $x \equiv \binom{x}{y} (mod\ m_i)$

组合数对任意数取模

- 将模数p质因数分解，将$x \equiv \binom{x}{y} (mod\ p)$打开成$n$个同余方程组：
$x \equiv \binom{x}{y} (mod\ m_i)$
- 最后将同余方程组用孙子定理合并

组合数对任意数取模

- 将模数p质因数分解，将$x \equiv \binom{x}{y} (mod\ p)$打开成$n$个同余方程组：
  $x \equiv \binom{x}{y} (mod\ m_i)$
- 最后将同余方程组用孙子定理合并
- 注意计算$\binom{x}{y} (mod\ m_i)$时，因为m_i可能不是质数，所以要把阶乘打开

组合数对任意数取模

- 将模数p质因数分解，将$x \equiv \binom{x}{y} (mod\ p)$打开成$n$个同余方程组：$x \equiv \binom{x}{y} (mod\ m_i)$
- 最后将同余方程组用孙子定理合并
- 注意计算$\binom{x}{y} (mod\ m_i)$时，因为m_i可能不是质数，所以要把阶乘打开
- $x! = 1 * 2 * 3 * ... * p * (p+1) * (p+2) * ... * (2p) * ...$，每$p$位分一组递归处理

组合数对任意数取模

- 将模数p质因数分解，将$x \equiv \binom{x}{y} (mod\ p)$打开成$n$个同余方程组：
  $x \equiv \binom{x}{y} (mod\ m_i)$
- 最后将同余方程组用孙子定理合并
- 注意计算$\binom{x}{y} (mod\ m_i)$时，因为m_i可能不是质数，所以要把阶乘打开
- $x! = 1 * 2 * 3 * ... * p * (p+1) * (p+2) * ... * (2p) * ...$，每$p$位分一组递归处理
- 时间复杂度$O(lognlogp)$

组合数对任意数取模

- 将模数p质因数分解，将$x \equiv \binom{x}{y} (mod\ p)$打开成$n$个同余方程组：$x \equiv \binom{x}{y} (mod\ m_i)$
- 最后将同余方程组用孙子定理合并
- 注意计算$\binom{x}{y} (mod\ m_i)$时，因为m_i可能不是质数，所以要把阶乘打开
- $x! = 1*2*3*...*p*(p+1)*(p+2)*...*(2p)*...$，每$p$位分一组递归处理
- 时间复杂度$O(lognlogp)$
- 例题：Codevs 礼物

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了
- 我的做法是将其转化为一个仅在对角线上有值的矩阵

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了
- 我的做法是将其转化为一个仅在对角线上有值的矩阵
- 当消第i列的时，在第$i-n$行中找出该列的最大值，交换这两行

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了
- 我的做法是将其转化为一个仅在对角线上有值的矩阵
- 当消第i列的时，在第$i-n$行中找出该列的最大值，交换这两行
- 将第i行的第i个元素消成1，并将其他列的元素消成0

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了
- 我的做法是将其转化为一个仅在对角线上有值的矩阵
- 当消第i列的时，在第$i-n$行中找出该列的最大值，交换这两行
- 将第i行的第i个元素消成1，并将其他列的元素消成0
- n次之后矩阵就只有对角线的地方有值了

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了
- 我的做法是将其转化为一个仅在对角线上有值的矩阵
- 当消第i列的时，在第$i-n$行中找出该列的最大值，交换这两行
- 将第i行的第i个元素消成1，并将其他列的元素消成0
- n次之后矩阵就只有对角线的地方有值了
- 注意：不要忘记第$n+1$列的元素

解方程组

- 网上有很多的解法都是转化成上三角矩阵，我这里就不说了
- 我的做法是将其转化为一个仅在对角线上有值的矩阵
- 当消第i列的时，在第$i-n$行中找出该列的最大值，交换这两行
- 将第i行的第i个元素消成1，并将其他列的元素消成0
- n次之后矩阵就只有对角线的地方有值了
- 注意：不要忘记第$n+1$列的元素
- 例题：Bzoj1013 球形空间产生器

线性基

- 又称极大线性无关组，名字高大上

线性基

- 又称极大线性无关组，名字高大上
- 其实线性基就是一个集合S，满足对于S的任意子集奇异或和不为0

线性基

- 又称极大线性无关组，名字高大上
- 其实线性基就是一个集合S，满足对于S的任意子集奇异或和不为0
- 利用$a\hat{\ }b = c \Leftrightarrow a\hat{\ }c = b$,所以按二进制位从高到低构造

线性基

- 又称极大线性无关组，名字高大上
- 其实线性基就是一个集合S，满足对于S的任意子集奇异或和不为0
- 利用$a\hat{}b = c \Leftrightarrow a\hat{}c = b$,所以按二进制位从高到低构造
- 如果欲将x插入到线性基P中，x最高位为i，如果P_i有值，那么令$x\hat{}= P_i$

线性基

- 又称极大线性无关组，名字高大上
- 其实线性基就是一个集合S，满足对于S的任意子集奇异或和不为0
- 利用$a\hat{}b = c \Leftrightarrow a\hat{}c = b$,所以按二进制位从高到低构造
- 如果欲将x插入到线性基P中，x最高位为i，如果P_i有值，那么令$x\hat{}= P_i$
- 继续，直到P_i没有值或者$x = 0$，(如果P_i没有值则令$P_i = x$)

线性基

- 又称极大线性无关组，名字高大上
- 其实线性基就是一个集合S，满足对于S的任意子集奇异或和不为0
- 利用$a\hat{}b = c \Leftrightarrow a\hat{}c = b$,所以按二进制位从高到低构造
- 如果欲将x插入到线性基P中，x最高位为i，如果P_i有值，那么令$x\hat{}= P_i$
- 继续，直到P_i没有值或者$x = 0$，(如果P_i没有值则令$P_i = x$)
- 例题：Hdu4939，Bzoj2640

Noip2011计算系数

- 给出多项式$(ax+by)^k$，求将其展开后$x^n y^m$的系数

Noip2011计算系数

- 给出多项式$(ax + by)^k$，求将其展开后$x^n y^m$的系数
- 二次项定理：$ans = a^n b^m * \binom{k}{m}$

Noip2011计算系数

- 给出多项式$(ax+by)^k$，求将其展开后$x^n y^m$的系数
- 二次项定理：$ans = a^n b^m * \binom{k}{m}$
- 组合数部分递推和逆元均可

Noip2012同余方程

- 给出a, b，求$ax \equiv 1(mod\ b)$的最小整数解

Noip2012同余方程

- 给出a, b，求$ax \equiv 1(mod\ b)$的最小整数解
- 拓展欧几里得算法

Noip2013转圈游戏

- n个人，编号$0-n-1$，每一轮第i个位置的人走到顺时针走m个位置，10^k轮后，原编号为x的人走到了哪里

Noip2013转圈游戏

- n个人，编号$0-n-1$，每一轮第i个位置的人走到顺时针走m个位置，10^k轮后，原编号为x的人走到了哪里
- $ans=(x+m*10^k)(mod\ n)$

Noip2013转圈游戏

- n个人，编号$0-n-1$，每一轮第i个位置的人走到顺时针走m个位置，10^k轮后，原编号为x的人走到了哪里
- $ans=(x+m*10^k)(mod\ n)$
- 快速幂就可以了

Noip2014解方程

- 求方程$\sum_{i=0}^{n} a_i x^i = 0$，求$x$在$[1, m]$内的所有整数解

Noip2014解方程

- 求方程$\sum\limits_{i=0}^{n} a_i x^i = 0$，求$x$在$[1, m]$内的所有整数解
- 模意义下的哈希：如果x_0是方程的一个整数解，那么x_0也是方程$\sum\limits_{i=0}^{n} a_i x^i \equiv 0 (mod\ \ p)$的整数解

Noip2014解方程

- 求方程$\sum_{i=0}^{n} a_i x^i = 0$，求$x$在$[1, m]$内的所有整数解
- 模意义下的哈希：如果x_0是方程的一个整数解，那么x_0也是方程$\sum_{i=0}^{n} a_i x^i \equiv 0 (mod\ p)$的整数解
- 我们可以选取几个质数p，求其在$[1, p]$的整数根即可

Noip2014解方程

- 求方程$\sum\limits_{i=0}^{n} a_i x^i = 0$，求$x$在$[1, m]$内的所有整数解
- 模意义下的哈希：如果x_0是方程的一个整数解，那么x_0也是方程$\sum\limits_{i=0}^{n} a_i x^i \equiv 0 (mod\ \ p)$的整数解
- 我们可以选取几个质数p，求其在$[1, p]$的整数根即可
- 注意整数根要多取几个(或者去网上找)，比如说$x = 2$不是$2x^2 = 0$的整数根，但却是$2x^2 \equiv 0 (mod\ \ 8)$的整数根