类欧几里得算 法

洪华敦

需要用的的一

关于欧几里得

类欧几里得算 法

一些简单的扩展

感谢

类欧几里得算法

洪华敦

浙江省绍兴市第一中学

关于欧几里得 算法

类欧几里得拿 法

一些简单的扩 展

-R 541

- [a] 若表达式a为真,则[a] = 1, 否则[a] = 0
- 例如[1+1=2]=1
- % 取模
- $S_d(n) = \sum_{x=1}^n x^d$
- | = | = | = | 下取整

欧几里得算法

洪华敦

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算

一些简单的扩 展

需要用的的-此数学符号

关于欧几里得 算法

类欧几里得算

一些简单的扩展

eR 14

• 对于 $b \neq 0$,有gcd(a,b) = gcd(b,a%b)

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算法

一些简单的扩 展

需要用的的一

关于欧几里得 算法

类欧几里得 法

一些简单的扩展

展

• 为了更好地讲后面的东西, 我们来证明下欧几里得算法的复杂度

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

极

• 为了更好地讲后面的东西, 我们来证明下欧几里得算法的复杂度

考虑最坏情况, a≤2*b-1

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得 法

一些简单的扩展

//C

- 为了更好地讲后面的东西, 我们来证明下欧几里得算法的复杂度
- 考虑最坏情况, a≤2*b-1
- 那么有gcd(a,b) = gcd(b,a-b)

关于欧几里得 算法

类欧几里得算 法

一些简单的扩展

n8 261

- 为了更好地讲后面的东西, 我们来证明下欧几里得算法的复杂度
- 考虑最坏情况, a≤2*b-1
- 那么有gcd(a,b) = gcd(b,a-b)
- 将a, b设为比较接近的斐波那契数,那 $\Delta a = F_n$, $b = F_{n-1}$,则 有 $gcd(F_n, F_{n-1}) = gcd(F_{n-1}, F_{n-2})$

关于欧几里得 算法

类欧几里得算 法

一些简单的扩展

- 为了更好地讲后面的东西, 我们来证明下欧几里得算法的复杂度
- 考虑最坏情况, a≤2*b-1
- 那么有gcd(a,b) = gcd(b,a-b)
- 将a, b设为比较接近的斐波那契数,那 $\Delta a = F_n$, $b = F_{n-1}$,则 有 $gcd(F_n, F_{n-1}) = gcd(F_{n-1}, F_{n-2})$
- 显然n是O(log(F_n))的

复杂度证明

洪华敦

需要用的的-此数学符号

关于欧几里得 算法

类欧几里得 法

一些简单的扩展

版

• 为了更好地讲后面的东西, 我们来证明下欧几里得算法的复杂度

- 考虑最坏情况, a≤2*b-1
- 那么有gcd(a,b) = gcd(b,a-b)
- 将a, b设为比较接近的斐波那契数,那 $\Delta a = F_n$, $b = F_{n-1}$,则 有 $gcd(F_n, F_{n-1}) = gcd(F_{n-1}, F_{n-2})$
- 显然n是O(log(F_n))的
- 所以这个算法的复杂度是O(log(a))的

需要用的的一

关于欧几里得算法

类欧几里得算

一些简单的扩 展

雲栗用的的-

关于欧几里得

类欧几里得算

一些简单的扩 展

. .

• 我们来看这样一个问题

关于欧几里得算法

类欧几里得算 法

一些简单的扩展

成油

- 我们来看这样一个问题
- $\sum_{d=1}^{n} (-1)^{\lfloor \sqrt{d*r*d} \rfloor}$

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的-此粉学符号

关于欧几里?

类欧几里得算 法

一些简单的扩展

nR 161

• 显然只要知道有多少 $\left[\sqrt{d*r*d}\right]$ 是奇数就行了

类欧几里得算 法

一些简单的扩展

感谚

- 显然只要知道有多少 $\left[\sqrt{d*r*d}\right]$ 是奇数就行了
- 于是就变成了求 $\sum_{d=1}^{n} \left\lfloor \frac{d*\sqrt{r}}{2} \right\rfloor$ %2

关于欧几里? 算法

类欧几里得算 法

一些简单的扩展

感谚

- 显然只要知道有多少 $\left[\sqrt{d*r*d} \right]$ 是奇数就行了
- 于是就变成了求 $\sum_{d=1}^{n} \left| \frac{d*\sqrt{r}}{2} \right| \%2$
- 考虑这个问题的一般形式 $\sum_{i=0}^{n} \left\lfloor \frac{ax+b}{c} \right\rfloor$

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

关于欧几里得

类欧几里得算 注

一些简单的扩 展

感谢

• 有一种传统的几何做法

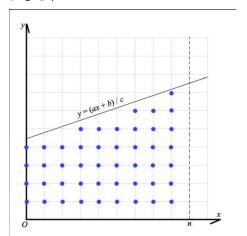
需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 有一种传统的几何做法
- 我们要求的是这个



需要用的的一 些数学符号

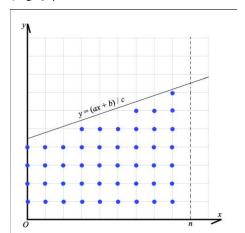
关于欧几里得 算法

类欧几里得算 法

一些简单的扩展

感谢

- 有一种传统的几何做法
- 我们要求的是这个



• (PS:图片选自其他课件)

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的一 此数学符号

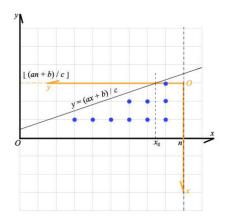
关于欧几里得

类欧几里得算 法

一些简单的扩展

感谢

• 我们先把下面的整块矩形直接计算掉



需要用的的一 此数学符号

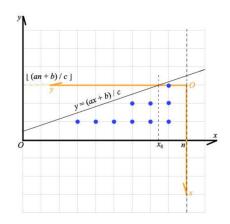
关于欧几里得

类欧几里得算 法

一些简单的抗

感谢

• 我们先把下面的整块矩形直接计算掉



• 我们以点 $(n, \lfloor \frac{a*n+b}{c} \rfloor)$ 为原点重建坐标轴

需要用的的一 些数学符号

关于欧几里得算法

类欧几里得算 注

一些简单的扩 展

需要用的的-此数学符号

关于欧几里得

类欧几里得算 法

一些简单的扩展

• 然后重新推导下这个直线的解析式

需要用的的-此粉学符号

关于欧几里征

类欧几里得算 法

一些简单的扩展

-R 141

- 然后重新推导下这个直线的解析式
- 显然斜率是变成了倒数了的, 所以a与c互换了

类欧几里得算

- 然后重新推导下这个直线的解析式
- 显然斜率是变成了倒数了的, 所以a与c互换了
- 可以发现复杂度和计算gcd类似,都是O(logn)

需要用的的一

关于欧几里? 算法

类欧几里得算 法

一些简单的扩展

感讲

- 然后重新推导下这个直线的解析式
- 显然斜率是变成了倒数了的, 所以a与c互换了
- 可以发现复杂度和计算gcd类似,都是O(logn)
- 这种做法虽然直观,但是有很大的局限性,下面介绍另一种做法

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的-

关于欧几里得

类欧几里得算 法

一些简单的扩 展

咸油

首先,如果a>= c,设d = [a]则有:

类欧几里得算 法

一些简单的扩 展

咸油

- 首先, 如果a>= c, 设d = | = | 则有:
- $\sum_{x=0}^{n} \left\lfloor \frac{ax+b}{c} \right\rfloor = \sum_{x=0}^{n} \left\lfloor \frac{(a\%c)x+b}{c} \right\rfloor + d * x$

一些简单的扩展

- 首先, 如果a >= c, 设d = [a]则有:
- $\sum_{x=0}^{n} \left\lfloor \frac{ax+b}{c} \right\rfloor = \sum_{x=0}^{n} \left\lfloor \frac{(a\%c)x+b}{c} \right\rfloor + d*x$

•
$$=d * S_1(n) + \sum_{x=0}^n \left\lfloor \frac{(a\%c)x+b}{c} \right\rfloor$$

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的-此粉学符号

类欧几里得算 法

一些简单的扩 展

成油

• 同理, 如果有b >= c, 设 $d = \left\lfloor \frac{b}{c} \right\rfloor$ 则有:

类欧几里得算 法

一些简单的扩 展

- 同理, 如果有b >= c, 设 $d = \left\lfloor \frac{b}{c} \right\rfloor$ 则有:
- $\sum_{x=0}^{n} \left\lfloor \frac{ax+b}{c} \right\rfloor = \sum_{x=0}^{n} \left\lfloor \frac{ax+(b\%c)}{c} \right\rfloor + d$

关于欧几里得 算法

类欧几里得算 法

- 同理, 如果有b >= c, 设 $d = \left\lfloor \frac{b}{c} \right\rfloor$ 则有:
- $\sum_{x=0}^{n} \left\lfloor \frac{ax+b}{c} \right\rfloor = \sum_{x=0}^{n} \left\lfloor \frac{ax+(b\%c)}{c} \right\rfloor + d$
- $=d * S_0(n) + \sum_{x=0}^n \left\lfloor \frac{ax + (b\%c)}{c} \right\rfloor$

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的一

关于欧几里得

类欧几里得算 注

一些简单的扩 展

eR 141

• 经过上面两个计算后, 我们有a < c, b < c

需要用的的-此数学符号

关于欧几里? 算法

类欧几里得算 法

一些简单的扩展

eR 161

- 经过上面两个计算后, 我们有a < c, b < c
- 于是我们的问题规模从(a, c, b, n)变成了(a%c, c, b%c, n)

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

类欧几里得算 法

一些简单的扩 展

•
$$\diamondsuit M = \left\lfloor \frac{an+b}{c} \right\rfloor$$

一些简单的扩 展

- $\diamondsuit M = \left| \frac{an+b}{c} \right|$
- \emptyset $\lesssim \sum_{x=0}^{n} \sum_{y=0}^{M} [y < \lfloor \frac{ax+b}{c} \rfloor]$

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- $\diamondsuit M = \lfloor \frac{an+b}{c} \rfloor$
- \emptyset $\preceq = \sum_{x=0}^{n} \sum_{y=0}^{M} [y < \lfloor \frac{ax+b}{c} \rfloor]$
- 我们来化简下这个式子 $y < \left\lfloor \frac{ax+b}{c} \right\rfloor$

一些简单的扩 展

•
$$\diamondsuit M = \left\lfloor \frac{an+b}{c} \right\rfloor$$

•
$$\emptyset$$
 $\preceq = \sum_{x=0}^{n} \sum_{y=0}^{M} [y < \lfloor \frac{ax+b}{c} \rfloor]$

- 我们来化简下这个式子 $y < \left\lfloor \frac{ax+b}{c} \right\rfloor$
- $c * (y + 1) \le a * x + b$

一些简单的**扩**展

- $\diamondsuit M = \left\lfloor \frac{an+b}{c} \right\rfloor$
- 我们来化简下这个式子 $y < \left\lfloor \frac{ax+b}{c} \right\rfloor$
- $c * (y + 1) \le a * x + b$
- $c * y + c b \le a * x$

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

•
$$\diamondsuit M = \lfloor \frac{an+b}{c} \rfloor$$

- 我们来化简下这个式子 $y < \left\lfloor \frac{ax+b}{c} \right\rfloor$
- $c * (y + 1) \le a * x + b$
- $c * y + c b \le a * x$
- c * y + c b 1 < a * x

类欧几里得算 法

一些简单的扩 展

感谢

•
$$\diamondsuit M = \left\lfloor \frac{an+b}{c} \right\rfloor$$

•
$$\emptyset$$
 $\lesssim \sum_{x=0}^{n} \sum_{y=0}^{M} [y < \lfloor \frac{ax+b}{c} \rfloor]$

• 我们来化简下这个式子 $y < \left\lfloor \frac{ax+b}{c} \right\rfloor$

•
$$c * (y + 1) \le a * x + b$$

•
$$c * y + c - b \le a * x$$

•
$$c * y + c - b - 1 < a * x$$

•
$$X > \left\lfloor \frac{c * y + c - b - 1}{a} \right\rfloor$$

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 注

一些简单的扩 展

关于欧几里得 6 注

类欧几里得算 法

一些简单的扩 展

成油

• 所以原式=
$$\sum_{y=0}^{M} \sum_{x=0}^{n} [x > \left\lfloor \frac{c*y+c-b-1}{a} \right\rfloor]$$

一些简单的扩 展

• 所以原式=
$$\sum_{y=0}^{M} \sum_{x=0}^{n} [x > \left\lfloor \frac{c*y+c-b-1}{a} \right\rfloor]$$

• =
$$\sum_{y=0}^{M} n - \left\lfloor \frac{c*y+c-b-1}{a} \right\rfloor$$

一些简单的扩 展

• 所以原式=
$$\sum_{y=0}^{M} \sum_{x=0}^{n} [x > \left| \frac{c*y+c-b-1}{a} \right|]$$

•
$$=\sum_{y=0}^{M} n - \left| \frac{c*y+c-b-1}{a} \right|$$

• =
$$n * (M + 1) - \sum_{y=0}^{M} \left\lfloor \frac{c * y + c - b - 1}{a} \right\rfloor$$

需要用的的-此数学符号

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

-R: 541

• 所以原式=
$$\sum_{y=0}^{M} \sum_{x=0}^{n} [x > \left| \frac{c*y+c-b-1}{a} \right|]$$

•
$$=\sum_{y=0}^{M} n - \left| \frac{c*y+c-b-1}{a} \right|$$

•
$$= n * (M + 1) - \sum_{y=0}^{M} \left\lfloor \frac{c * y + c - b - 1}{a} \right\rfloor$$

• 于是我们的问题规模从(a%c, c, b%c, n)变成了(c, a%c, c - b%c - 1, M)

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩展

类欧几里得算 法

一些简单的扩 展

感谢

• 经过上面一番转化,我们的问题规模从(a,c,b,n)变成了(c,a%c,c-b%c-1,M)

复杂度

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 经过上面一番转化, 我们的问题规模从(a, c, b, n)变成了(c, a%c, c b%c 1, M)
- 显然当a为0时,我们可以很轻松地计算答案

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 经过上面一番转化, 我们的问题规模从(a, c, b, n)变成了(c, a%c, c b%c 1, M)
- 显然当a为0时, 我们可以很轻松地计算答案
- 所以问题规模每次都从(a,c)变成(c,a%c)

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩展

- 经过上面一番转化, 我们的问题规模从(a, c, b, n)变成了(c, a%c, c b%c 1, M)
- 显然当a为0时, 我们可以很轻松地计算答案
- 所以问题规模每次都从(a, c)变成(c, a%c)
- 这和计算gcd的时间复杂度是一样的

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩展

r8 261

- 经过上面一番转化, 我们的问题规模从(a, c, b, n)变成了(c, a%c, c b%c 1, M)
- 显然当a为0时, 我们可以很轻松地计算答案
- 所以问题规模每次都从(a, c)变成(c, a%c)
- 这和计算gcd的时间复杂度是一样的
- 所以时间复杂度是O(log(a))

需要用的的一 些数学符号

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

- 经过上面一番转化, 我们的问题规模从(a, c, b, n)变成了(c, a%c, c b%c 1, M)
- 显然当a为0时, 我们可以很轻松地计算答案
- 所以问题规模每次都从(a, c)变成(c, a%c)
- 这和计算gcd的时间复杂度是一样的
- 所以时间复杂度是O(log(a))
- 看上去这种做法更加的麻烦,但是这种方法是可以扩展 到高维情况的



真的吗?那爽死了

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算法

一些简单的扩 展

需要用的的-些粉学符号

关于欧几里?

类欧几里得算 法

一些简单的扩 展

成油

简化题意:

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

成油

- 简化题意:
- 给定N, A, B, K, L, 计算:

$$\sum_{x=1}^{n} {\binom{\left\lfloor \frac{Ax}{b} \right\rfloor}{K+1}} {\binom{n-x}{L}}$$

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

感谢

- 简化题意:
- 给定N, A, B, K, L, 计算:

$$\sum_{x=1}^{n} {\binom{\left\lfloor \frac{Ax}{b} \right\rfloor}{K+1}} {\binom{n-x}{L}}$$

• $N, A, B \le 10^{18}$, $K, L \le 10$

需要用的的-些数学符号

关于欧几里? 算法

类欧几里得事 法

一些简单的扩 展

- 简化题意:
- 给定N, A, B, K, L, 计算:

$$\sum_{x=1}^{n} \binom{\left\lfloor \frac{Ax}{b} \right\rfloor}{K+1} \binom{n-x}{L}$$

- $N, A, B \le 10^{18}, K, L \le 10$
- 答案对一个大质数取模

需要用的的一 些数学符号

关于欧几里得 链法

类欧几里得算 法

一些简单的扩 展

需要用的的一

关于欧几里?

类欧几里得算 法

一些简单的扩 展

.

• 首先我们知道, $\binom{x}{k}$ 是一个关于x的k次多项式

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

D 1/1

- 首先我们知道, $\binom{x}{k}$ 是一个关于x的k次多项式
- $i \chi^{\Gamma} \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right) = \sum_{i=0}^{K+1} c_i * \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right)^i$

类欧几里得算 法

一些简单的扩 展

- 首先我们知道, $\binom{x}{t}$ 是一个关于x的k次多项式
- $i \chi \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right) = \sum_{i=0}^{K+1} c_i * \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right)^i$
- $\mathop{\mathfrak{P}}\nolimits \binom{n-x}{L} = \sum_{i=0}^L d_i * x^i$

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

成油

- 首先我们知道, $\binom{x}{t}$ 是一个关于x的k次多项式
- $i \chi \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right) = \sum_{i=0}^{K+1} c_i * \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right)^i$
- $\mathfrak{F}\binom{n-x}{L} = \sum_{i=0}^{L} d_i * x^i$

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

D 1/1

- 首先我们知道, $\binom{x}{t}$ 是一个关于x的k次多项式
- $i \chi \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right) = \sum_{i=0}^{K+1} c_i * \left(\left\lfloor \frac{Ax}{b} \right\rfloor \right)^i$
- $\mathfrak{F}\binom{n-x}{l} = \sum_{i=0}^{L} d_i * x^i$
- 那么原式= $\sum_{x=1}^{n} \sum_{i=0}^{L} d_i * x^i \sum_{j=0}^{K+1} c_j * (\lfloor \frac{Ax}{b} \rfloor)^j$
- \$\xi\frac{\xi}{\infty}\sum_{i=0}^{L}\sum_{j=0}^{K+1}d_i*c_j\sum_{x=1}^nx^i*\left(\left[\frac{Ax}{b}\right]\right)^j\$

一些简单的扩

- 首先我们知道, (x)是一个关于x的k次多项式
- $\mathfrak{F}(\left\lfloor \frac{Ax}{b} \right\rfloor) = \sum_{i=0}^{K+1} c_i * (\left\lfloor \frac{Ax}{b} \right\rfloor)^i$
- $\mathfrak{F}(n-x) = \sum_{i=0}^{L} d_i * x^i$
- 那么原式= $\sum_{i=0}^{n} \sum_{i=0}^{L} d_i * x^i \sum_{i=0}^{K+1} c_i * (|\frac{Ax}{b}|)^i$
- 等于 $\sum_{i=0}^{L} \sum_{i=0}^{K+1} d_i * c_j \sum_{x=1}^{n} x^i * (|\frac{Ax}{b}|)^j$
- d:和c:我们可以预处理用拉格朗日插值公式或者高斯消元算出来. 那 么问题就变成了求后面那些东西

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的一 此数学符号

关于欧几里得

类欧几里得算

一些简单的扩 展

感谢

• 我们令f(a,c,b,i,j)表示 $\sum_{x=0}^{n} x^{i} * (\lfloor \frac{a*x+b}{c} \rfloor)^{j}$

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 我们令f(a,c,b,i,j)表示 $\sum_{x=0}^{n} x^{i} * (\lfloor \frac{a*x+b}{c} \rfloor)^{j}$
- 首先我们可以用二项式定理, 让a, b < c

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 我们令f(a,c,b,i,j)表示 $\sum_{x=0}^{n} x^{i} * (\lfloor \frac{a*x+b}{c} \rfloor)^{j}$
- 首先我们可以用二项式定理, 让a,b < c
- $\diamondsuit M = \left\lfloor \frac{a*n+b}{c} \right\rfloor$

需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得美 法

一些简单的扩 展

- 我们令f(a,c,b,i,j)表示 $\sum_{x=0}^{n} x^{i} * (\left| \frac{a*x+b}{c} \right|)^{j}$
- 首先我们可以用二项式定理, 让a,b < c
- $\diamondsuit M = \left\lfloor \frac{a*n+b}{c} \right\rfloor$

$$\sum_{x=0}^{n} x^{U} * \left(\left\lfloor \frac{a * x + b}{c} \right\rfloor \right)^{V}$$

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

感谢

- 我们令f(a, c, b, i, j)表示 $\sum_{x=0}^{n} x^{i} * (|\frac{a*x+b}{c}|)^{j}$
- 首先我们可以用二项式定理, 让a, b < c
- $\diamondsuit M = \left\lfloor \frac{a*n+b}{c} \right\rfloor$

$$\sum_{x=0}^{n} x^{U} * \left(\left\lfloor \frac{a * x + b}{c} \right\rfloor \right)^{V}$$

•

$$= \sum_{x=0}^{n} x^{U} * \sum_{y=0}^{M} [y < \left\lfloor \frac{a * x + b}{c} \right\rfloor] ((y+1)^{V} - y^{V})$$

需要用的的一 些数学符号

关于欧几里得 60 注

类欧几里得算 法

一些简单的扩 展

感谢

• 我们令
$$f(a, c, b, i, j)$$
表示 $\sum_{x=0}^{n} x^{i} * (|\frac{a*x+b}{c}|)^{j}$

•
$$\diamondsuit M = \left\lfloor \frac{a*n+b}{c} \right\rfloor$$

$$\sum_{x=0}^{n} x^{U} * \left(\left\lfloor \frac{a * x + b}{c} \right\rfloor \right)^{V}$$

•

$$= \sum_{x=0}^{n} x^{U} * \sum_{y=0}^{M} [y < \left\lfloor \frac{a * x + b}{c} \right\rfloor] ((y+1)^{V} - y^{V})$$

•

$$= \sum_{y=0}^{M} ((y+1)^{V} - y^{V}) \sum_{x=0}^{n} x^{U} * [x > \left\lfloor \frac{c * y + c - b - 1}{a} \right\rfloor]$$

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的-此数学符号

关于欧几里得

类欧几里得算 法

一些简单的扩 展

感谢

• 我们可以把 $(y+1)^U - y^U$ 二项式展开

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

成油

- 我们可以把 $(y+1)^U y^U$ 二项式展开
- 原式等于

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

咸油

- 我们可以把 $(y+1)^U y^U$ 二项式展开
- 原式等于
- = $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * \sum_{x=0}^{n} x^U * [x > \lfloor \frac{c*y+c-b-1}{a} \rfloor]$

需要用的的-此数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

-8: ×41

- 我们可以把 $(y+1)^U y^U$ 二项式展开
- 原式等于
- = $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * \sum_{x=0}^{n} x^U * [x > \lfloor \frac{c*y+c-b-1}{a} \rfloor]$
- $\bullet = \sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^{i} * \left(S_{U}(n) S_{U}\left(\left\lfloor \frac{c*y + c b 1}{a} \right\rfloor \right) \right)$

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 我们可以把 $(y+1)^U y^U$ 二项式展开
- 原式等于

• =
$$\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^{i} * \sum_{x=0}^{n} x^{U} * [x > \left\lfloor \frac{c*y+c-b-1}{a} \right\rfloor]$$

• =
$$\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * (S_U(n) - S_U(\lfloor \frac{c*y+c-b-1}{a} \rfloor))$$

• =
$$\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^{i} * S_{U}(n) - \sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^{i} * S_{U}(\left|\frac{c*y+c-b-1}{a}\right|)$$

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

- 我们可以把 $(y+1)^U y^U$ 二项式展开
- 原式等于
- = $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^{i} * \sum_{x=0}^{n} x^{U} * [x > \left\lfloor \frac{c*y+c-b-1}{a} \right\rfloor]$
- = $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * (S_U(n) S_U(\lfloor \frac{c*y+c-b-1}{a} \rfloor))$
- = $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * S_U(n) \sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * S_U(\left\lfloor \frac{c*y+c-b-1}{a} \right\rfloor)$
- 前者等于 $\sum_{i=0}^{V-1} {V \choose i} S_i(M) * S_U(n)$,于是问题就变成了求后者

需要用的的一 些数学符号

关于欧几里得 链法

类欧几里得算 法

一些简单的扩 展

需要用的的一

关于欧几里征

类欧几里得算 法

一些简单的扩 展

-R 141

• 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

成油

- 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式

类欧几里得 法

一些简单的扩 展

成油

- 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式
- 数组a可以用高斯消元预处理,或者直接带伯努利数进去

需要用的的-

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

咸油

- 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式
- $i \xi S_d(x) = \sum_{i=0}^{d+1} a[d][i] * x^i$
- 数组a可以用高斯消元预处理,或者直接带伯努利数进去
- 原式等于 $\sum_{i=0}^{V-1} \binom{V}{i} \sum_{y=0}^{M} y^i * \sum_{j=0}^{U+1} a[U][j] * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^j$

类欧几里得算 法

一些简单的扩 展

感讲

- 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式
- 数组a可以用高斯消元预处理,或者直接带伯努利数进去
- 原式等于 $\sum_{i=0}^{V-1} \binom{V}{i} \sum_{y=0}^{M} y^i * \sum_{j=0}^{U+1} a[U][j] * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^j$
- \$\xi \tau_{i=0}^{V-1} \sum_{j=0}^{U+1} a[U][j] * \binom{v}{i} \sum_{y=0}^{M} y^{i} * (\binom{c**y+c-b-1}{a}\binom{j}{i})^{j}

需要用的的-

关于欧几里?

类欧几里得算 法

一些简单的扩 展

感讲

- 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式
- 数组a可以用高斯消元预处理,或者直接带伯努利数进去
- 原式等于 $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * \sum_{j=0}^{U+1} a[U][j] * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^j$
- \(\xi + \sum_{i=0}^{V-1} \sum_{j=0}^{U+1} a[U][j] * \binom{\varphi}{i} \sum_{y=0}^{M} y^{i} * \left(\left[\frac{c*y+c-b-1}{a} \right] \right)^{j}
- 于是我们再次完成了a, c的互换,这样递归下去只有O(log(A))层

需要用的的一

关于欧几里? 算法

类欧几里得算 法

一些简单的扩 展

- 我们可以发现 $S_d(x)$ 是一个关于x的d+1次多项式
- $i \xi S_d(x) = \sum_{i=0}^{d+1} a[d][i] * x^i$
- 数组a可以用高斯消元预处理,或者直接带伯努利数进去
- \emptyset 系式等于 $\sum_{i=0}^{V-1} {V \choose i} \sum_{y=0}^{M} y^i * \sum_{j=0}^{U+1} a[U][j] * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^j$
- 于是我们再次完成了a, c的互换,这样递归下去只有O(log(A))层
- 我们可以对于每个(a, b, c, n)求出对于所有(i, j)的答案,具体方法就 是递归下去,然后用返回的那些答案更新

需要用的的一 些数学符号

关于欧几里得 算法

类欧几里得算 法

一些简单的扩 展

需要用的的-

关于欧几里征

类欧几里得算 法

一些简单的扩 展

-R 141

• 然而裸着直接做好像是 $O((K+L)^4 logn)$ 的,我们可以优化一下

需要用的的一

关于欧几里得

类欧几里得算 法

一些简单的扩 展

感讲

- 然而裸着直接做好像是O((K + L)⁴ logn)的,我们可以优化一下
- 对于每一个i, 计算出 $\sum_{j=0}^{U+1} a[U][j] \sum_{y=0}^{M} y^{i} * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^{j}$, 这个是 $(K+L)^{3}$ 的

需要用的的一

关于欧几里?

类欧几里得算 法

一些简单的扩 展

感说

- 然而裸着直接做好像是O((K+L)⁴logn)的,我们可以优化一下
- 对于每一个i, 计算出 $\sum_{j=0}^{U+1} a[U][j] \sum_{y=0}^{M} y^{i} * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^{j}$, 这个是 $(K+L)^{3}$ 的
- 然后对于每个V直接计算即可

需要用的的一

关于欧几里?

类欧几里得算 法

一些简单的扩 展

- 然而裸着直接做好像是O((K + L)⁴ logn)的, 我们可以优化一下
- 对于每一个i, 计算出 $\sum_{j=0}^{U+1} a[U][j] \sum_{y=0}^{M} y^{i} * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^{j}$, 这个是 $(K+L)^{3}$ 的
- 然后对于每个V直接计算即可
- 时间复杂度O((K+L)³logn)

一些简单的扩

- 然而裸着直接做好像是O((K+L)⁴logn)的,我们可以优 化一下
- 对于每一个i,计算出 $\sum_{j=0}^{U+1} a[U][j] \sum_{y=0}^{M} y^{j} * (\lfloor \frac{c*y+c-b-1}{a} \rfloor)^{j}$, 这个是 $(K + L)^3$ 的
- 然后对于每个V直接计算即可
- 时间复杂度O((K+L)³logn)
- 当然如果你用FFT的话复杂度会继续降下来



需要用的的一 此数学符号

关于欧几里得 算法

类欧几里得算

一些简单的扩展

需要用的的-此粉学符号

关于欧几里征

类欧几里得^身

一些简单的扩展 展

感谢

• 感谢CCF给了我这次交流的机会

需要用的的一

关于欧几里? 算法

类欧几里得美法

一些简单的扩展

- 感谢CCF给了我这次交流的机会
- 感谢绍兴一中的任之洲和叶菁菁同学为本文审稿

需要用的的-

关于欧几里? 算法

类欧几里得到法

一些简单的扩展

- · 感谢CCF给了我这次交流的机会
- 感谢绍兴一中的任之洲和叶菁菁同学为本文审稿
- 感谢清华大学的张恒捷和杭州学军中学的金策在这个知识上对我的启发