类欧几里得算法

引入

考虑这样一个等式$\sum_{d=1}^{n}(-1)^{\lfloor\sqrt{d*r*d}\rfloor}$

可以化为$\sum_{d=1}^{n}(\lfloor\frac{d*\sqrt{r}}{2}\rfloor\&1)$,更一般的形式为$\sum_{i=0}^{n}\lfloor\frac{ax+b}{c}\rfloor$

可以通俗考虑直线下的整数坐标点，计算的复杂度是$O(logn)$的。

如果有$a\geq c$，设$d=\lfloor\frac{a}{c}\rfloor$则有:

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

同理如果有$b\geq c$，设$d=\lfloor\frac{b}{c}\rfloor$则有:

$\sum_{x=0}^{n}\lfloor\frac{ax+b}{c}\rfloor=S_0(n)*d+\sum_{x=0}^{n}\lfloor\frac{ax+(b\%c)}{c}\rfloor$

经过两次计算后问题的规模就会从(a,b,c,n)变成$(a\%c,b\%c,c,n)$

接下来我们只需要考虑$a<c,b<c$的情况，考虑原式的贡献变为$\sum_{i=0}^{n}\sum_{j=0}^{\lfloor\frac{ai+b}{c}-1\rfloor}1$

交换i,j的求和算子，强制用n限制i的上界：$\sum_{j=0}^{\lfloor\frac{ai+b}{c}-1\rfloor}\sum_{i=0}^{n}[j<\lfloor\frac{ai+b}{c}\rfloor]$

然后把向下取整的符号拿掉：$j<\lfloor\frac{ai+b}{c}\rfloor\iff j+1\leq\frac{ai+b}{c}$

计算i可以化为$\lfloor\frac{jc+c-b-1}{a}\rfloor<i$，这一步的意义在于可以把$i$消去，令$m=\frac{an+b}{c}$代入可以得到

$$
\begin{aligned}
f(a,b,c,n)&=\sum_{j=0}^{m-1}\sum_{i=0}^{n}[i>\lfloor\frac{jc+c-b-1}{a}\rfloor]=\sum_{j=0}^{m-1}(n-\lfloor\frac{jc+c-b-1}{a}\rfloor)\\
&=nm-f(c,c-b-1,a,m-1)
\end{aligned}
$$

这就是基础类欧几里得算法最终的推式，复杂度$O(logn)$

基本结论

设$f(a,b,c,n)=\sum_{i=0}^{n}\lfloor\frac{ai+b}{c}\rfloor$，当$a\geq c$或$b\geq c$时有:

①$f(a,b,c,n)=\frac{n(n+1)}{2}\lfloor\frac{a}{c}\rfloor+(n+1)\lfloor\frac{b}{c}\rfloor+f(a\%c,b\%c,c,n)$

②$f(a,b,c,n)=nm-f(c,c-b-1,a,m-1)$, 其中$m=\frac{an+b}{c}$

扩展

考虑两种变种求和式：

$g(a,b,c,n)=\sum_{i=0}^{n} i\lfloor \frac{ai+b}{c} \rfloor$

$h(a,b,c,n)=\sum_{i=0}^{n} \lfloor \frac{ai+b}{c} \rfloor^2$

统一设$m=\lfloor \frac{an+b}{c} \rfloor, t=\lfloor \frac{jc+c-b-1}{a} \rfloor$

$g(a,b,c,n)=g(a\%c,b\%c,c,n)+\lfloor \frac{a}{c} \rfloor \frac{n(n+1)(2n+1)}{6}+\frac{b}{c}\frac{n(n+1)}{2}$

$$h(a,b,c,n)=h(a\%b,b\%c,c,n)+2\lfloor \frac{b}{c} \rfloor f(a\%c,b\%c,c,n)+2\lfloor \frac{a}{c} \rfloor g(a\%c,b\%c,c,n)$$
$$+\lfloor \frac{a}{c} \rfloor^2 \frac{n(n+1)(2n+1)}{6}+\lfloor \frac{b}{c} \rfloor^2 (n+1)+\lfloor \frac{a}{c} \rfloor \lfloor \frac{b}{c} \rfloor n(n+1)$$

对于$a<c$且$b<c$的情况有
$g(a,b,c,n)=\frac{1}{2}[mn(n+1)-h(c,c-b-1,a,m-1)-f(c,c-b-1,a,m-1)]$

$h(a,b,c,n)=nm(m+1)-2g(c,c-b-1,a,m-1)-2f(c,c-b-1,a,m-1)-f(a,b,c,n)$

推导略。

可以发现三个函数式是相互关联的，因此要考虑整体递归以减少运算量。

参考文档

OI-wiki

类欧几里得算法_洪华敦