类欧几里得算法

算法思想

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

其中 a,b,c,n 为常数，我们需要一个 $O(logn)$ 的算法。

如果 $a≥c$ 或者 $b≥c$ □我们可以将 a,b 对 c 取模来化简问题：

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

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

$={\frac {n(n+1)} {2}}\lfloor {\frac a c} \rfloor+(n+1)\lfloor {\frac b c} \rfloor+\sum_{i=0}^{n}\lfloor {\frac {(a\ mod\ c)i+(b\ mod\ c)} {c}} \rfloor$

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

这样我们就将前两个参数控制到一定比第三个参数小的形式了。

我们有 $\sum_{i=0}^{n}\lfloor {\frac {ai+b} {c}} \rfloor=\sum_{i=0}^{n}\sum_{j=0}^{\lfloor {\frac {ai+b} {c}} \rfloor-1}1$

然后我们交换和号：

$\sum_{j=0}^{\lfloor {\frac {an+b} {c}} \rfloor-1}\sum_{i=0}^{n}[j<\lfloor {\frac {ai+b} {c}} \rfloor]$

对于里面的式子，我们可以变换一下：

$j<\lfloor {\frac {ai+b} {c}} \rfloor \Leftrightarrow j+1≤ \lfloor {\frac {ai+b} {c}} \rfloor \Leftrightarrow j+1≤{\frac {ai+b} {c}} \Leftrightarrow jc+c≤ai+b \Leftrightarrow jc+c-b-1□ai \Leftrightarrow \lfloor {\frac {jc+c-b-1} {a}} \rfloor □i$

这样我们设 $m= \lfloor {\frac {an+b} {c}} \rfloor$

原式变为□ $f(a,b,c,n)=\sum_{j=0}^{m-1}\sum_{i=0}^{n}[i□\lfloor {\frac {jc+c-b-1} {a}} \rfloor]=\sum_{j=0}^{m-1}(n-\lfloor {\frac {jc+c-b-1} {a}} \rfloor)=nm-f(c,c-b-1,a,m-1)$ □然后第一个参数又比第三个大了，就一直取模这样，类似于求最大公约数。

算法实现

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;
```

```
const ll mod=998244353,base=233;

ll Hash(int a,int b,int c,int n) {
    return (((((ll)a*base%mod)+b)*base%mod+c)*base%mod+n)%mod;
}
unordered_map<ll,int> F;

ll f(int a,int b,int c,int n) {
    if(!a) return (ll)b/c*(n+1)%mod;
    ll tmp=Hash(a,b,c,n);
    if(F.find(tmp)!=F.end()) return F[tmp];
    if(a>=c||b>=c) return F[tmp]=(((ll)n*(n+1)/2%mod*(a/c)%mod+((ll)n+1)*(b/c)%mod)%mod+f(a%c,b%c,c,n))%mod;
    int m=((ll)a*n+b)/c;
    return F[tmp]=((ll)n*m%mod-f(c,c-b-1,a,m-1)+mod)%mod;
}
int n,a,b,c;
int main() {
    int t;
    scanf("%d",&t);
    while(t--) {
        scanf("%d %d %d %d",&n,&a,&b,&c);
        printf("%lld\n",f(a,b,c,n));
    }
    return 0;
}
```

代码练习

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

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

当 $a==0$ 时， $g(a,b,c,n)=(n+1){\lfloor {\frac b c} \rfloor}^{2}$ ${\frac {n(n+1)} {2}}\lfloor {\frac b c} \rfloor$

当 $a≥c$ 或 $b≥c$ 时， $g(a,b,c,n)=\sum_{i=0}^{n}{(\lfloor {\frac {i(a\ mod\ c)+b\ mod\ c} {c}} \rfloor +i \lfloor {\frac a c} \rfloor + \lfloor {\frac b c} \rfloor)}^{2}$

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

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

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

$+{\frac {n(n+1)(2n+1)} 6}{\lfloor {\frac a c} \rfloor}^{2}+n(n+1)\lfloor {\frac a c} \rfloor \lfloor {\frac b c} \rfloor+(n+1){\lfloor {\frac b c} \rfloor}^{2}$

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

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

当 $a□c$ 且 $b□c$ 时，仍设 $m=\lfloor {\frac {an+b} c} \rfloor$

$g(a,b,c,n)=2\sum_{i=0}^{n}\sum_{j=1}^{\lfloor {\frac {ai+b} c} \rfloor}j-\sum_{i=0}^{n}\lfloor {\frac {ai+b} c} \rfloor$

$=-f(a,b,c,n)+2\sum_{i=0}^{n}\sum_{j=1}^{m}j[j≤{\lfloor {\frac {ai+b} c} \rfloor}]$

$=-f(a,b,c,n)+2\sum_{i=0}^{n}\sum_{j=1}^{m-1}(j+1)[(j+1)c□ai+b+1]$

$=-f(a,b,c,n)+2\sum_{j=0}^{m-1}(j+1)\sum_{i=0}^{n}[i>{\lfloor {\frac {jc+c-b-1} {a}} \rfloor}]$

$=-f(a,b,c,n)+2\sum_{j=0}^{m-1}(j+1)(n-{\lfloor {\frac {jc+c-b-1} {a}} \rfloor})$

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

$h(a,b,c,n)=\sum_{i=0}^{n}i\sum_{j=1}^{m}[j≤\lfloor {\frac {ai+b} {c}} \rfloor]$

$=\sum_{i=0}^{n}i\sum_{j=0}^{m-1}[(j+1)c□ai+b+1]$

$=\sum_{j=0}^{m-1}\sum_{i=0}^{n}i[i□{\frac {jc+c-b-1} {a}}]$

$=\sum_{j=0}^{m-1}({\frac {n(n+1)} {2}}-\sum_{i=0}^{n}i[i≤{\lfloor {\frac {jc+c-b-1} {a}} \rfloor}])$

$=\sum_{j=0}^{m-1}({\frac {n(n+1)} {2}}-{\frac {{\lfloor {\frac {jc+c-b-1} {a}} \rfloor}(\lfloor {\frac {jc+c-b-1} {a}} \rfloor+1)} {2}})$

$={\frac {\sum_{j=0}^{m-1}n(n+1)-\sum_{j=0}^{m-1}{\lfloor {\frac {jc+c-b-1} {a}} \rfloor}^{2}-\sum_{j=0}^{m-1}{\lfloor {\frac {jc+c-b-1} {a}} \rfloor}} {2}}$

$={\frac 1 2}[mn(n+1)-g(c,c-b-1,a,m-1)-f(c,c-b-1,a,m-1)]$

注意负数那里会导致计算错误，比如 ${\frac {-1} 2}={\frac {1} 2}=0$

所以在实际计算中，要规避掉参数为负的情况，这个具体情况具体分析。

```
#include <bits/stdc++.h>
using namespace std;
typedef long long ll;

const ll mod=998244353,inv2=(mod+1)>>1,inv3=332748118;
```

```
struct Ans {
    ll f,s,t;
}ans,now,las;
Ans getans(int a,int b,int c,int n) {
    if(!a) {
        now.f=(ll)b/c*(n+1)%mod;
        now.s=(ll)(b/c)*(b/c)%mod*(n+1)%mod;
        now.t=((ll)n+1)*n/2%mod*(b/c)%mod;
    }
    else if(a>=c||b>=c) {
        las=getans(a%c,b%c,c,n);
        now.f=(((ll)n*(n+1)/2%mod*(a/c)%mod+((ll)n+1)*(b/c)%mod)%mod+las.f)%mod;
        ll tmp1=((las.s+2ll*(a/c)%mod*las.t%mod)%mod+2ll*(b/c)%mod*las.f%mod)%mod;
        ll tmp2=(((ll)n*(n+1)%mod*(2ll*n+1)%mod*inv2%mod*inv3%mod*(a/c)%mod*(a/c)%mod+(ll)n*(n+1)%mod*(a/c)%mod*(b/c)%mod)%mod+(ll)(n+1)*(b/c)%mod*(b/c)%mod)%mod;
        now.s=(tmp1+tmp2)%mod;
        now.t=(((ll)n*(n+1)/2%mod*(b/c)%mod+((ll)n+1)*n%mod*inv2%mod*((ll)n*2+1)%mod*inv3%mod*(a/c)%mod)%mod+las.t)%mod;
    }else {
        ll m=((ll)a*n+b)/c;
        las=getans(c,c-b-1,a,m-1);
        now.f=((ll)n*m%mod-las.f+mod)%mod;
        now.s=((ll)n*m%mod*(m+1)%mod-now.f+mod-2ll*las.t%mod+mod-2ll*las.f%mod+mod)%mod;
        now.t=((ll)m*n%mod*(n+1)%mod-las.f+mod-las.s+mod)%mod*inv2%mod;
    }
    return now;
}
int n,a,b,c;
int main() {
    int t;
    scanf("%d",&t);
    while(t--) {
        scanf("%d %d %d %d",&n,&a,&b,&c);
        ans=getans(a,b,c,n);
        printf("%lld %lld %lld\n",ans.f,ans.s,ans.t);
        //printf("%lld %lld %lld\n",f(a,b,c,n),g(a,b,c,n),h(a,b,c,n));
    }
    return 0;
}
/*
10
7 7 7 7
8 0 10 4
8 2 4 3
3 4 5 3
0 3 10 1
```

```
1 0 0 7
3 9 10 1
8 5 5 4
5 4 0 9
10 4 4 9
*/
```

ps 这里一起计算是因为，单独记忆化 t 了…