

复健复健!

1 dp (打牌)

1.1 斜率优化

2 字符串

2.1 kmp (看猫片)

2.2 ac自动机 (树土看猫片)

2.3 后缀数组 (sa)

2.4 后缀自动机

2.5 回文自动机 (PAM)

3 数学数学

3.1 欧拉筛法 ($prime, \varphi, \mu$) , 线性求逆元。

3.2 gcd,lcm

3.3 费马小/欧拉定理

3.4 exgcd

3.5 容斥原理

3.6 Lucas定理

3.7 Catalan数

3.8 范德蒙德卷积

3.9 FFT

3.10 NTT

3.11 多项式全家桶

3.12 min_25/洲阁筛

样例输入 #1

样例输出 #1

样例输入 #2

样例输出 #2

提示

3.13 杜教筛

3.14 辛普森积分

3.15 Miller_Rabin && Pollard_Rho

3.16 五边形数, 整数拆分

4 数 (树) 据结构

4.1 树状数组

4.2 线段树

4.3 并查集

4.4 ST表

4.5 平衡树

4.6 LCT

4.7 bitset

5 图论 (吐子)

5.1 LCA (最近公共祖先)

5.2 最小生成树

5.3 最短路

5.4 树链剖分

5.5 网络最大流

5.6 最小费用最大流

上下界网络流

概述

无源汇上下界可行流

有源汇上下界可行流

有源汇上下界最大流

有源汇上下界最小流

5.7 二分图最大权完美匹配

5.8 一般图最大匹配 (带花树)

复健复健！

1 dp (打牌)

1.1 斜率优化

[P3195 \[HNOI2008\]玩具装箱](#)

$$\begin{aligned}f_n &= f_m + (n - m + s_n - s_m - L)^2 \\f_n &= f_m + (T_n - T_m - L)^2 \\f_n - T_n^2 - L^2 + 2LT_n + 2T_nT_m &= f_m + T_m^2 + 2LT_m\end{aligned}$$

点为 $(T_m, f_m + T_m^2 + 2LT_m)$,

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 50000
using namespace std;

int n, L, a[maxn + 5];
LL sum[maxn + 5], f[maxn + 5], t[maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

LL get_y(int x){
    return f[x] + t[x] * t[x] + 2ll * L * t[x];
}
```

```

double get_k(int x, int y){
    return (double)(get_y(x) - get_y(y)) / (t[x] - t[y]);
}

int st[maxn + 5], l = 1, r = 1;
int main(){
    //freopen("in", "r", stdin);
    read(n); read(L); L++;
    For(i, 1, n) read(a[i]), sum[i] = sum[i - 1] + a[i], t[i] = sum[i] + i;
    For(i, 1, n){
        while(l < r && get_k(st[l], st[l + 1]) < 2ll * t[i]) l++;
        f[i] = f[st[l]] + (t[i] - t[st[l]] - L) * (t[i] - t[st[l]] - L);
        while(l < r && get_k(st[r - 1], st[r]) > get_k(st[r], i)) r--;
        st[++r] = i;
    }
    printf("%lld\n", f[n]);
    return 0;
}

```

2 字符串

2.1 kmp (看猫片)

[P3375 【模板】KMP字符串匹配](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
using namespace std;

int n, m, to[maxn + 5];
char s1[maxn + 5], s2[maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;

```

```

    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int main(){
    //freopen("in", "r", stdin);
    scanf("%s %s", s1 + 1, s2 + 1);
    n = strlen(s1 + 1); m = strlen(s2 + 1);
    int now = 0;
    For(i, 2, m){
        while(now && s2[now + 1] != s2[i]) now = to[now];
        if(s2[now + 1] == s2[i]) to[i] = ++now;
    }
    now = 0;
    For(i, 1, n){
        while(now && s2[now + 1] != s1[i]) now = to[now];
        if(s2[now + 1] == s1[i]) now++;
        if(now == m) printf("%d\n", i - m + 1);
    }
    For(i, 1, m) printf("%d ", to[i]);
    return 0;
}

```

2.2 ac自动机（树上看猫片）

[P3808 【模板】AC 自动机（简单版）](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
using namespace std;

int n, tot = 0, pos[maxn + 5], as = 0;
char s[maxn + 5];
struct Node{
    int ch[30], to, ok;
} tr[maxn + 5];

template <class T>
void read(T &x){

```

```

char ch;
bool ok;
for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
if(ok) x = -x;
}

int q[maxn + 5], l = 1, r = 0;
void sol(){
    For(i, 0, 25) if(tr[0].ch[i]) q[++r] = tr[0].ch[i];
    while(l <= r){
        int now = q[l++];
        For(i, 0, 25){
            int to = tr[now].ch[i];
            if(to) tr[to].to = tr[tr[now].to].ch[i], q[++r] = to;
            else tr[now].ch[i] = tr[tr[now].to].ch[i];
        }
    }
}

int main(){
    // freopen("in", "r", stdin);
    read(n);
    For(i, 1, n){
        scanf("%s", s + 1);
        int len = strlen(s + 1), now = 0;
        For(j, 1, len){
            int x = s[j] - 'a';
            if(!tr[now].ch[x]) tr[now].ch[x] = ++tot;
            now = tr[now].ch[x];
        }
        pos[i] = now;
    }
    sol();
    scanf("%s", s + 1);
    int len = strlen(s + 1), now = 0;
    For(i, 1, len){
        now = tr[now].ch[s[i] - 'a'];
        tr[now].ok = 1;
    }
    For(i, r, 1) tr[tr[q[i]].to].ok |= tr[q[i]].ok;
    For(i, 1, n) as += tr[pos[i]].ok;
    printf("%d\n", as);
    return 0;
}

```

2.3 后缀数组 (sa)

[P3809 【模板】后缀排序](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)

```

```

#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
using namespace std;

int n, m;
char s[maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

struct SA {
    int n, m, rk[maxn + 5], sum[maxn + 5], to[maxn + 5], nto[maxn + 5], rk2[maxn + 5], cnt, ncnt;
    int height[maxn + 5];
    struct RMQ {
        int n, a[maxn + 5], pre[maxn + 5][25], suf[maxn + 5][25];
        void init(int *_a, int _n) {
            n = _n;
            For(i, 1, n) a[i] = _a[i];
            For(i, 1, n) {
                suf[i][0] = i;
                For(o, 1, 20) {
                    suf[i][o] = suf[i][o - 1];
                    int to = i - (1 << (o - 1));
                    if(to <= 0) continue;
                    if(a[suf[to][o - 1]] < a[suf[i][o]]) suf[i][o] = suf[to][o - 1];
                }
            }
            Rof(i, n, 1) {
                pre[i][0] = i;
                For(o, 1, 20) {
                    pre[i][o] = pre[i][o - 1];
                    int to = i + (1 << (o - 1));
                    if(to > n) continue;
                    if(a[pre[to][o - 1]] < a[pre[i][o]]) pre[i][o] = pre[to][o - 1];
                }
            }
        }
    };
};

```

```

    }
}
}
int que(int l, int r) {
    if(l > r) swap(l, r); r--;
    assert(l <= r);
    int o = 0;
    while((l << o) <= r - l + 1) o++; o--;
    return min(a[pre[l][o]], a[suf[r][o]]);
}
} rmq;
void init(char *s) {
    n = strlen(s + 1);
    m = max(n, 200);
    For(i, 1, m) sum[i] = 0;
    For(i, 1, n) ++sum[to[i] = s[i] + 1];
    For(i, 1, m) sum[i] += sum[i - 1];
    For(i, 1, n) rk[sum[to[i]]--] = i;
    for(int i = 1; i <= n; i <= 1) {
        cnt = 0;
        For(j, n - i + 1, n) rk2[++cnt] = j;
        For(j, 1, n) if(rk[j] > i) rk2[++cnt] = rk[j] - i;
        For(j, 1, m) sum[j] = 0;
        For(j, 1, n) sum[to[j]]++;
        For(j, 1, m) sum[j] += sum[j - 1];
        For(j, n, 1) rk[sum[to[rk2[j]]]--] = rk2[j];
        ncnt = 0;
        For(j, 1, n) {
            int tem1 = rk[j - 1] + i > n ? 0 : to[rk[j - 1] + i], tem2 =
rk[j] + i > n ? 0 : to[rk[j] + i];
            if(j == 1 || to[rk[j - 1]] != to[rk[j]] || tem1 != tem2) ncnt++;
            nto[rk[j]] = ncnt;
        }
        For(j, 1, n) to[j] = nto[j];
        if(ncnt == n) break;
    }
    s[n + 1] = '$';
    int now = 0;
    For(i, 1, n) {
        if(now) now--;
        if(to[i] == n) continue;
        while(s[i + now] == s[rk[to[i] + 1] + now]) now++;
        height[to[i]] = now;
    }
    rmq.init(height, n - 1);
}
int lcp(int p1, int p2) { return rmq.que(to[p1], to[p2]); }
} sa;

```

2.4 后缀自动机

[P6139 【模板】广义后缀自动机 \(广义 SAM\)](#)

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
using namespace std;

int n, m, tot = 0, ch[maxn + 5][26];
LL as = 0;
char s[maxn + 5];
struct Node{
    int len, ch[26], fa;
} poi[2 * maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int tot1 = 1;
int ins(int x, int las){
    int p = las, np;
    if(poi[p].ch[x] && poi[p].len + 1 == poi[poi[p].ch[x]].len) return poi[p].ch[x];
    else np = ++tot;
    poi[np].len = poi[p].len + 1;
    while(p && !poi[p].ch[x]) poi[p].ch[x] = np, p = poi[p].fa;
    if(!p) poi[np].fa = 1;
    else {
        int q = poi[p].ch[x];
        if(poi[p].len + 1 == poi[q].len) poi[np].fa = q;
        else {
            int nq = ++tot; poi[nq] = poi[q];
            poi[nq].len = poi[p].len + 1;
            poi[np].fa = poi[q].fa = nq;
        }
    }
}
```



```

        while(p && poi[p].ch[x] == q) poi[p].ch[x] = nq, p = poi[p].fa;
    }
}
return np;
}

int to[maxn + 5];
void sol(){
    queue<int> q;
    q.push(0);
    to[0] = 1;
    while(q.size()){
        int u = q.front(); q.pop();
        For(i, 0, 25){
            int v = ch[u][i];
            if(!v) continue;
            to[v] = ins(i, to[u]);
            q.push(v);
        }
    }
}

int main(){
    //freopen("in", "r", stdin);
    read(n);
    For(i, 1, n){
        scanf("%s", s + 1); m = strlen(s + 1);
        int now = 0;
        For(j, 1, m){
            int x = s[j] - 'a';
            if(!ch[now][x]) ch[now][x] = ++tot;
            now = ch[now][x];
        }
    }
    sol();
    For(i, 1, tot1) as += poi[i].len - poi[poi[i].fa].len;
    printf("%lld\n", as);
    return 0;
}

```

2.5 回文自动机 (PAM)

[P5496 【模板】回文自动机 \(PAM\)](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back

```

```

#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 500000
using namespace std;

int n;
char s[maxn + 5];
int poi_tot = 0;
struct Poi{
    int len, cnt;
    Poi *ch[26], *to;
} poi[maxn + 5];
Poi *get_new(){
    poi[++poi_tot] = {};
    return &poi[poi_tot];
}
struct Tree{
    Poi *Rt[2];
    Tree(){
        Rt[0] = get_new();
        Rt[1] = get_new();
        Rt[0]->to = Rt[1]->to = Rt[1];
        Rt[1]->len = -1;
    }
    Poi *get_o(Poi *rt, int o){
        if(!rt->ch[o]) return Rt[0];
        return rt->ch[o];
    }
    Poi* ins(Poi *las, int pos){
        int o = s[pos] - 'a';
        while(s[pos - las->len - 1] != s[pos]) las = las->to;
        if(las->ch[o]) return las->ch[o];
        else{
            Poi *now = get_new(), *pre = las->to;
            now->len = las->len + 2;
            while(s[pos - pre->len - 1] != s[pos]) pre = pre->to;
            pre = get_o(pre, o);
            now->to = pre;
            now->cnt = pre->cnt + 1;
            las->ch[o] = now;
            return now;
        }
    }
} tr;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());

```

```

    if(ok) x = -x;
}

int main(){
    // freopen("in.txt", "r", stdin);
    scanf("%s", s + 1); n = strlen(s + 1);
    int res = 0;
    Poi *las = tr.Rt[0];
    For(i, 1, n) tr.ins(las, i);
    return 0;
}

```

3 数学数学

3.1 欧拉筛法 ($prime, \varphi, \mu$) , 线性求逆元。

$$n = \prod_{i=1}^m p_i^{a_i}$$

$$\varphi_n = n \sum_{p_i} \left(1 - \frac{1}{p_i}\right)$$

$$\mu_n = \begin{cases} 0 & \exists i \in [1, m] s.t. a_i \geq 2 \\ (-1)^m & otherwise \end{cases}$$

$$n^{-1} = -\left\lfloor \frac{p}{n} \right\rfloor (p \% n)^{-1}$$

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 100000
using namespace std;

int n, pri[maxn + 5], cnt = 0, vis[maxn + 5], phi[maxn + 5], mu[maxn + 5],
inv[maxn + 5];

template <class T>
void read(T &x){
    char ch;

```

```

bool ok;
for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
if(ok) x = -x;
}

int P = 998244353
int main(){
    //freopen("in", "r", stdin);
    read(n);
    inv[1] = 1;
    For(i, 2, n){
        inv[i] = mod - P / i * inv[P % i] % mod;
        if(!vis[i]){
            pri[++cnt] = i;
            phi[i] = i - 1;
            mu[i] = -1;
        }
        for(int j = 1; j <= cnt && i * pri[j] <= n; ++j){
            vis[i * pri[j]] = 1;
            if(i % pri[j] == 0){
                phi[i * pri[j]] = phi[i] * pri[j];
                break;
            }
            else{
                phi[i * pri[j]] = phi[i] * (pri[j] - 1);
                mu[i * pri[j]] = -mu[i];
            }
        }
    }
    return 0;
}

```

3.2 gcd,lcm

$$\gcd(a, b) = \gcd(b, a \% b)$$

$$\text{lcm}(a, b) = \frac{a \cdot b}{\gcd(a, b)}$$

3.3 费马小/欧拉定理

若 $\gcd(a, m) = 1$, 有:

$$a^{\varphi(m)} \equiv 1 \pmod{m}$$

3.4 exgcd

方程 $ax + by = c$, 若有:

$$bx' + (a \% b)y' = c$$

那么:

$$\begin{cases} x = y' \\ y = x' - \lfloor \frac{a}{b} \rfloor y' \end{cases}$$

3.5 容斥原理

绝大部分容斥的本质是这个式子：

$$\sum_{i=0}^n \binom{n}{i} (-1)^i = [n = 0]$$

有变形：

$$[n = k] \iff [n - k = 0]$$

$$[n = k] = \sum_{i=0}^{n-k} \binom{n-k}{i} (-1)^i$$

容斥原理相关有，莫比乌斯反演：

$$f(n) = \sum_{d|n} g(d) \iff g(n) = \sum_{d|n} \mu\left(\frac{n}{d}\right) f(d)$$

二项式反演：

$$f(n) = \sum_{i=0}^n \binom{n}{i} g(i) \iff g(n) = \sum_{i=0}^n \binom{n}{i} (-1)^{n-i} f(i)$$

min-max容斥：

$$\min\{S\} = \sum_{T \in S} (-1)^{|T|+1} \max\{T\}$$

$$\max\{S\} = \sum_{T \in S} (-1)^{|T|+1} \min\{T\}$$

3.6 Lucas定理

对于素数p，有：

$$\binom{n}{k} \equiv \binom{\lfloor \frac{n}{p} \rfloor}{\lfloor \frac{k}{p} \rfloor} \cdot \binom{n \% p}{k \% p} \pmod{p}$$

3.7 Catalan数

$$H_n = \frac{\binom{2n}{n}}{n+1} = \binom{2n}{n} - \binom{2n}{n-1}$$

可以表示：

1. 有2n个人排成一行进入剧场。入场费 5 元。其中只有n个人有一张 5 元钞票，另外n人只有 10 元钞票，剧院无其它钞票，问有多少种方法使得只要有 10 元的人买票，售票处就有 5 元的钞票找零？
2. 一位大城市的律师在她住所以北n个街区和以东n个街区处工作。每天她走2n个街区去上班。如果他从不穿越（但可以碰到）从家到办公室的对角线，那么有多少条可能的道路？
3. 在圆上选择2n个点，将这些点成对连接起来使得所得到的n条线段不相交的方法数。
4. n个结点可构造多少个不同的二叉树？

5. 一个栈（无穷大）的进栈序列为1, 2, 3, ..., n, 有多少个不同的出栈序列。

3.8 范德蒙德卷积

$$\sum_{i=1}^k \binom{n}{i} \binom{m}{k-i} = \binom{n+m}{k}$$

3.9 FFT

n项多项式 $f_n(x) = \sum_{i=0}^{n-1} a_i x^i$, 其快速傅里叶变换为:

$$b_k = \sum_{i=0}^{n-1} a_i (\omega_n^k)^i$$

对其傅里叶逆变换为:

$$c_k = \sum_{i=0}^{n-1} b_i (-\omega_n^k)^i = \sum_{i=0}^{n-1} (\omega_n^{-k})^i \sum_{j=0}^{n-1} a_j (\omega_n^i)^j = n a_k$$

[P3803 【模板】多项式乘法 \(FFT\)](#)

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
using namespace std;

int n, m;
double pi = acos(-1);

struct Com{
    double a, b;
    Com(){}
    Com(double _a, double _b){
        a = _a;
        b = _b;
    }
}
a[4 * maxn + 5], b[4 * maxn + 5];
Com operator + (Com &x, Com &y){return (Com){x.a + y.a, x.b + y.b};}
Com operator - (Com &x, Com &y){return (Com){x.a - y.a, x.b - y.b};}
```

```

Com operator * (Com &x, Com &y){return (Com){x.a * y.a - x.b * y.b, x.a * y.b +
x.b * y.a};}

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int lim, to[4 * maxn + 5];
void fft_pre(int x){
    int l = 0; lim = 1;
    while(lim < x) lim <<= 1, l++;
    for(i, 0, lim - 1) to[i] = (to[i >> 1] >> 1) + ((i & 1) << (l - 1));
}

void fft(Com *x, int gi){
    for(i, 0, lim - 1) if(i < to[i]) swap(x[i], x[to[i]]);
    for(int i = 2; i <= lim; i <<= 1){
        Com w = Com(1.0 * cos(2.0 * pi / i), gi * sin(2.0 * pi / i));
        for(int j = 0; j < lim; j += i){
            int ii = i >> 1;
            Com wi = Com(1, 0);
            for(p, j, j + ii - 1){
                Com tem1 = x[p], tem2 = wi * x[p + ii];
                x[p] = tem1 + tem2;
                x[p + ii] = tem1 - tem2;
                wi = wi * w;
            }
        }
    }
}

int main(){
    // freopen("in", "r", stdin);
    read(n); read(m); n++; m++;
    for(i, 0, n - 1) scanf("%lf", &a[i].a);
    for(i, 0, m - 1) scanf("%lf", &b[i].a);
    fft_pre(n + m - 1);
    fft(a, 1); fft(b, 1);
    // cout << lim << endl;
    // for(i, 0, lim - 1) cout << b[i].a << " " << b[i].b << endl;
    for(i, 0, lim - 1) a[i] = a[i] * b[i];
    fft(a, -1);
    for(i, 0, n + m - 2) printf("%lld ", (LL)(a[i].a / lim + 0.5));
    return 0;
}

```

3.10 NTT

将 $g = 3^{\frac{p-1}{n}}$ 作为 ω_n^1 , 其中 g 为 p 的一个原根, p 通常取998244353。

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
#define mod 998244353
using namespace std;

int n, m, g = 3, ig, a[4 * maxn + 5], b[4 * maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int fp(int x, int y){
    int asi = 1;
    while(y){
        if(y & 1) asi = 1ll * asi * x % mod;
        x = 1ll * x * x % mod;
        y >>= 1;
    }
    return asi;
}

int lim, lim_inv, to[4 * maxn + 5];
void ntt_pre(int x){
    int l = 0; lim = 1;
    while(lim < x) lim <<= 1, l++;
    For(i, 0, lim - 1) to[i] = (to[i >> 1] >> 1) + ((i & 1) << (l - 1));
    lim_inv = fp(lim, mod - 2);
}

void ntt(int *x, int gi){
    For(i, 0, lim - 1) if(i < to[i]) swap(x[i], x[to[i]]);
```



```

for(int i = 2; i <= lim; i <= 1){
    int w = fp(gi, (mod - 1) / i), ii = i >> 1;
    for(int j = 0; j < lim; j += i){
        int wi = 1;
        For(p, j, j + ii - 1){
            int tem1 = x[p], tem2 = 111 * wi * x[p + ii] % mod;
            x[p] = (tem1 + tem2) % mod;
            x[p + ii] = (tem1 - tem2 + mod) % mod;
            wi = 111 * wi * w % mod;
        }
    }
}

int main(){
    //freopen("in", "r", stdin);
    ig = fp(g, mod - 2);
    read(n); read(m); n++; m++;
    For(i, 0, n - 1) read(a[i]);
    For(i, 0, m - 1) read(b[i]);
    ntt_pre(n + m - 1);
    ntt(a, g); ntt(b, g);
    For(i, 0, lim - 1) a[i] = 111 * a[i] * b[i] % mod;
    ntt(a, ig);
    For(i, 0, n + m - 2) printf("%d ", 111 * a[i] * lim_inv % mod);
    return 0;
}

```

3.11 多项式全家桶

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
using namespace std;

template<class T>
T fp(T x, int y){
    T asi = 1;
    while(y){
        if(y & 1) asi *= x;
        x *= x;
    }
}

```

```

        y >>= 1;
    }
    return asi;
}

template<int P>
struct Mint{
    int x;
    Mint(){x = 0;}
    Mint(int _x) : x{norm(_x % P)}{}
    int norm(int _x){
        if(_x < 0) _x += P;
        if(_x >= P) _x -= P;
        return _x;
    }
    Mint inv(){return fp(*this, P - 2);}
    Mint operator - (){return {P - x};}
    Mint& operator += (Mint _x) &{
        x = norm(x + _x.x);
        return *this;
    }
    Mint& operator -= (Mint _x) &{
        x = norm(x - _x.x);
        return *this;
    }
    Mint& operator *= (Mint _x) &{
        x = 1ll * x * _x.x % P;
        return *this;
    }
    Mint& operator /= (Mint _x) &{return *this *= _x.inv();}
    friend Mint operator + (Mint _x, Mint _y){return _x += _y;}
    friend Mint operator - (Mint _x, Mint _y){return _x -= _y;}
    friend Mint operator * (Mint _x, Mint _y){return _x *= _y;}
    friend Mint operator / (Mint _x, Mint _y){return _x /= _y;}
    friend bool operator == (Mint _x, Mint _y){return _x.x == _y.x;}
};

const int P = 998244353;
using Z = Mint<P>;

namespace NTT{
    Z g = 3;
    vector<int> rev;
    vector<Z> gpw{0, 1};
    void dft(vector<Z> &f){
        int lim = f.size();
        if(rev.size() != lim){
            rev.resize(lim);
            int k = __builtin_ctz(lim) - 1;
            For(i, 0, lim - 1) rev[i] = (rev[i >> 1] >> 1) | ((i & 1) << k);
        }
        For(i, 0, lim - 1) if(rev[i] < i) swap(f[rev[i]], f[i]);
        if(gpw.size() < lim){
            int k = __builtin_ctz(gpw.size());
            gpw.resize(lim);
            while((1 << k) < lim){

```

```

        Z w = fp(g, (P - 1) >> (k + 1));
        For(i, 1 << (k - 1), (1 << k) - 1){
            gpw[i << 1] = gpw[i];
            gpw[(i << 1) | 1] = gpw[i] * w;
        }
        k++;
    }
}

for(int i = 2; i <= lim; i <= 1){
    int ii = i >> 1;
    for(int j = 0; j < lim; j += i){
        For(p, 0, ii - 1){
            Z tem1 = f[j + p], tem2 = gpw[ii + p] * f[j + ii + p];
            f[j + p] = tem1 + tem2;
            f[j + ii + p] = tem1 - tem2;
        }
    }
}

}

void idft(vector<Z> &f){
    int lim = f.size();
    reverse(f.begin() + 1, f.end());
    dft(f);
    Z inv = Z(lim).inv();
    For(i, 0, lim - 1) f[i] *= inv;
}

}

struct Poly : vector<Z>{
    Poly(){}
    explicit Poly(int len) : vector<Z>(len){}
    explicit Poly(cst vector<Z> &f) : vector<Z>(f){}
    Poly(cst initializer_list<Z> &a) : std::vector<Z>(a){}

    template<class InputIt>
    explicit constexpr Poly(InputIt itl, InputIt itr) : std::vector<Z>(itl, itr)
    {}

    Poly resz(int len){
        Poly res = *this;
        res.resize(len);
        return res;
    }

    Poly shift(int len){
        if(len >= 0){
            Poly res = *this;
            res.insert(res.begin(), len, 0);
            return res;
        }
        else if(len + size() < 0) return Poly();
        return Poly(begin() - len, end());
    }

    Poly operator += (cst Poly &f){
        if(f.size() > size()) resize(f.size());

```

```

    int o = size() == f.size() ? size() : f.size();
    For(i, 0, o - 1) (*this)[i] += f[i];
    return *this;
}

Poly operator -= (cst Poly &f){
    if(f.size() > size()) resize(f.size());
    int o = size() == f.size() ? size() : f.size();
    For(i, 0, o - 1) (*this)[i] -= f[i];
    return *this;
}

Poly operator *= (Poly f){
    if(!size() || !f.size()) return {0};
    int len = size() + f.size() - 1, lim = 1;
    while(lim < len) lim <= 1;
    resize(lim);
    f.resize(lim);
    NTT::dft(*this); NTT::dft(f);
    For(i, 0, lim - 1) (*this)[i] *= f[i];
    NTT::idft(*this);
    return resz(len);
}

Poly operator *= (cst Z &x){
    For(i, 0, size() - 1) (*this)[i] *= x;
    return *this;
}

friend Poly operator + (Poly f, cst Poly &g){return f += g;}
friend Poly operator - (Poly f, cst Poly &g){return f -= g;}
friend Poly operator * (Poly f, cst Poly &g){return f *= g;}
friend Poly operator * (Poly f, cst Z &x){return f *= x;}
friend Poly operator * (cst Z &x, Poly f){return f *= x;}

Poly deri(){
    Poly res(size() - 1);
    For(i, 0, size() - 2) res[i] = (*this)[i + 1] * (i + 1);
    return res;
}

Poly inte(){
    Poly res(size() + 1);
    For(i, 1, size()) res[i] = (*this)[i - 1] / i;
    return res;
}

Poly inv(int mlim){
    Poly res(1), tem, tem2(1); res[0] = (*this)[0].inv(); tem2[0] = 2;
    int lim = 1;
    while(lim < mlim){
        lim <= 1;
        tem = *this;
        tem.resize(lim);
        res = (res * (tem2 - tem * res)).resz(lim);
    }
    return res.resz(mlim);
}

Poly ln(int mlim){return (inv(mlim) * deri()).inte().resz(mlim);}
Poly exp(int mlim){
    Poly res(1), tem, tem1(1); res[0] = 1; tem1[0] = 1;
    int lim = 1;
    while(lim < mlim){

```

```

        lim <= 1;
        tem = *this; tem.resize(lim);
        res = (res * (tem1 - res.ln(lim) + tem)).resz(lim);
    }
    return res.resz(mlim);
}

Poly sqrt(int mlim){
    Poly res(1), tem; res[0] = 1;
    int lim = 1;
    while(lim < mlim){
        lim <= 1;
        tem = *this; tem.resize(lim);
        res = (res - (res * res - tem) * ((Z)2 * res).inv(lim)).resz(lim);
    }
    return res.resz(mlim);
    // return ((Z){2}.inv() * ln(mlim)).exp(mlim).resz(mlim);
}

pair<Poly, Poly> divi(Poly f){
    if(f.size() > size()) return {{0}, *this};
    Poly tem = *this, temf = f;
    int lim = size() - f.size() + 1;
    reverse(temf.begin(), temf.end()); temf.resize(lim);
    reverse(tem.begin(), tem.end()); tem.resize(lim);
    Poly tt = temf.inv(lim);
    tem *= tt;
    tem.resize(lim);
    reverse(tem.begin(), tem.end());
    Poly res = *this - tem * f; res.resize(f.size() - 1);
    return {tem, res};
}

Poly pfp(int mlim, int k){
    Poly res = {1}, f = *this;
    while(k){
        if(k & 1) res = res * f;
        f *= f;
        k >>= 1;
    }
    return res;
    // return ((Z){k} * ln(mlim)).exp(mlim).resz(mlim);
}

vector<Z> eval(vector<Z> x) cst{
    vector<Poly> f(4 * x.size());
    function<void(int, int, int)> build = [&](int now, int l, int r){
        int mid = l + r >> 1;
        if(l == r){
            f[now] = {-x[l], 1};
            return;
        }
        f.pb({});
        build(now << 1, l, mid); build((now << 1) | 1, mid + 1, r);
        f[now] = f[now << 1] * f[(now << 1) | 1];
        return;
    };
    build(1, 0, x.size() - 1);
    vector<Z> res(x.size());

```

```

function<void(int, int, int, Poly)> sol = [&](int now, int l, int r, Poly
g){
    int mid = l + r >> 1;
    g = g.divi(f[now]).sec;
    if(l == r){
        res[l] = g[0];
        return;
    }
    sol(now << 1, l, mid, g); sol((now << 1) | 1, mid + 1, r, g);
};
sol(1, 0, x.size() - 1, *this);
return res;
}
};

namespace LinearRecurrence{
    Z sol(Poly p, Poly q, LL x){
        int lim = max(p.size(), q.size());
        p.resize(lim); q.resize(lim);
        while(x){
            Poly nq = q, np;
            for(int i = 1; i < lim; i += 2) nq[i] *= -1;
            np = p * nq;
            nq *= q;
            For(i, 0, lim - 1) p[i] = np[(i << 1) | (x & 1)];
            For(i, 0, lim - 1) q[i] = nq[i << 1];
            x >>= 1;
        }
        return p[0] / q[0];
    }
}

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int main(){
    // freopen("in.txt", "r", stdin);
    return 0;
}

```

3.12 min_25/洲阁筛

定义积性函数 $f(x)$, 且 $f(p^k) = p^k(p^k - 1)$ (p 是一个质数), 求

$$\sum_{i=1}^n f(i)$$

对 $10^9 + 7$ 取模。

样例输入 #1

10

样例输出 #1

263

样例输入 #2

1000000000

样例输出 #2

710164413

提示

$f(1) = 1, f(2) = 2, f(3) = 6, f(4) = 12, f(5) = 20;$
 $f(6) = 12, f(7) = 42, f(8) = 56, f(9) = 72, f(10) = 40.$

```
#include<bits/stdc++.h>
#define For(i, a, b) for(LL i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(LL i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
using namespace std;

template <class T>
void read(T& x) {
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
```

```

        if(ok) x = -x;
    }

#define P 1000000007
#define maxn 200000
#define maxm 600000

int fp(int x, int y) {
    int res = 1;
    while(y) {
        if(y & 1) res = 1ll * res * x % P;
        x = 1ll * x * x % P;
        y >>= 1;
    }
    return res;
}

int inv2 = fp(2, P - 2), inv6 = fp(6, P - 2);

namespace Min25 {
    LL n;
    int s[3][maxn + 5], pri[maxn + 5], cnt = 0, g[3][maxm + 5], vis[maxn + 5];

    int to[2][maxn + 5], tot = 0;
    LL id[maxm + 5];
    int To(LL x) {
        if(x <= maxn) return to[0][x];
        return to[1][n / x];
    }

    int get_f(LL x, int y) {
        if(pri[y] > x) return 1;
        int t = To(x);
        if(1ll * pri[y] * pri[y] > x) return ((1ll * g[2][t] - g[1][t]) - (1ll *
s[2][y - 1] - s[1][y - 1]) + 2ll * P + 1) % P;
        int res = 0;
        for(LL i = 1; i <= x; i *= pri[y])
            res = (res + 1ll * (i == 1 ? 1 : (1ll * i % P * (i % P) % P - i % P +
P) % P) * get_f(x / i, y + 1) % P) % P;
        return res;
    }

    void mian(LL _n) {
        n = _n;
        For(o, 0, 2) s[o][0] = 1;
        For(i, 2, maxn) {
            if(!vis[i]) {
                pri[++cnt] = i;
                int tmp = 1;
                For(o, 0, 2) {
                    s[o][cnt] = (s[o][cnt - 1] + tmp) % P;
                    tmp = 1ll * tmp * i % P;
                }
            }
            for(int j = 1; j <= cnt && i * pri[j] <= maxn; ++j) {
                vis[i * pri[j]] = 1;
            }
        }
    }
}

```



```

        if(i % pri[j] == 0) break;
    }
}
For(i, 1, n) {
    LL now = n / i;
    id[++tot] = now;
    if(now <= maxn) to[0][now] = tot;
    else to[1][n / now] = tot;
    g[0][tot] = now % P;
    g[1][tot] = 111 * now % P * ((now + 1) % P) % P * inv2 % P;
    g[2][tot] = 111 * now % P * ((now + 1) % P) % P * (211 * now % P + 1)
% P * inv6 % P;
    i = n / now;
}
For(o, 1, cnt) {
    For(i, 1, tot) {
        if(111 * pri[o] * pri[o] <= id[i]) {
            int tmp = 1;
            For(o1, 0, 2) {
                g[o1][i] = (g[o1][i] - 111 * (g[o1][To(id[i] / pri[o])]) -
s[o1][o - 1] + P) % P * tmp % P + P) % P;
                tmp = 111 * tmp * pri[o] % P;
            }
        }
        else break;
    }
}
printf("%d\n", get_f(n, 1));
}
}

LL n;
int main() {
    // freopen("in.txt", "r", stdin);
    read(n);
    Min25::mian(n);
}

```

```

#include<bits/stdc++.h>
#define For(i, a, b) for(LL i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(LL i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000

```

```

#define maxm 100000000011
using namespace std;

int pri[maxn + 5], cnt = 0, vis[maxn + 5];

class Sieve{
public:
    int to[maxn + 5], tot = 0, k_n[maxn + 5];
    LL N, sum_f_p[maxn + 5];
    map<int, int> id;
    vector<LL> F[maxn + 5], vF[maxn + 5], G[maxn + 5], vG[maxn + 5];
    LL get_f(LL x){
        //
    }
    LL get_g(LL x){
        //
    }
    Sieve(){}
    LL sol(LL _N){
        N = _N;
        tot = 0;
        For(i, 1, N){
            ++tot;
            id[to[tot] = N / i] = tot;
            i = N / (N / i);
        }
        int now = cnt;
        For(i, 1, tot){
            while(1ll * pri[now] * pri[now] > to[i]) now--;
            F[i].clear();
            F[i].resize(now + 5, 0);
            vF[i].clear();
            vF[i].resize(now + 5, 0);
            G[i].clear();
            G[i].resize(now + 5, 0);
            vG[i].clear();
            vG[i].resize(now + 5, 0);
            k_n[i] = now;
        }
        For(i, 1, cnt) sum_f_p[i] = sum_f_p[i - 1] + get_g(pri[i]);
        return get_F(1, 1);
    }
    LL get_sum_G(LL x){
        //
    }
    LL get_G(int n_id, int k){
        if(pri[k] > to[n_id]) return get_g(1);
        if(k == 1) return get_sum_G(to[n_id]);
        if(1ll * pri[k - 1] * pri[k - 1] > to[n_id]) return get_G(n_id,
k_n[n_id] + 1) - (sum_f_p[k - 1] - sum_f_p[k_n[n_id]]);
        if(vG[n_id][k]) return G[n_id][k];
        vG[n_id][k] = 1;
        LL asi = get_G(n_id, k - 1) - get_g(pri[k - 1]) * get_G(id[to[n_id] /
pri[k - 1]], k - 1);
        return G[n_id][k] = asi;
    }
}

```

```

LL get_F(int n_id, int k){
    if(1ll * pri[k] * pri[k] > to[n_id]) return get_G(n_id, k);
    if(vF[n_id][k]) return F[n_id][k];
    vF[n_id][k] = 1;
    LL now = 1, asi = 0;
    For(i, 0, inf){
        if(!(to[n_id] / now)) break;
        asi += get_f(now) * get_F(id[to[n_id] / now], k + 1);
        now *= pri[k];
    }
    return F[n_id][k] = asi;
}
} sieve;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int main(){
    // freopen("in.txt", "r", stdin);
    For(i, 2, maxn){
        if(!vis[i]) pri[++cnt] = i;
        for(int j = 1; j <= cnt && i * pri[j] <= maxn; ++j){
            vis[i * pri[j]] = 1;
            if(i % pri[j] == 0) break;
        }
    }
    return 0;
}

```

3.13 杜教筛

$$h = f \circ g$$

$$\sum_{i=1}^n h(i) = \sum_{i=1}^n \sum_{d|n} f(d) g\left(\frac{n}{d}\right) = \sum_{i=1}^n g(i) \sum_{j=1}^{\lfloor \frac{n}{i} \rfloor} f(j)$$

$$\sum_{i=1}^n f(i) = \sum_{i=1}^n h(i) - \sum_{i=2}^n g(i) \sum_{j=1}^{\lfloor \frac{n}{i} \rfloor} f(j)$$

```

#include<bits/stdc++.h>
#define For(i, a, b) for(LL i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(LL i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)

```

```

#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 30000000
#define maxm 3000000
using namespace std;

class Sieve{
public:
    int pri[maxm + 5], cnt = 0, miu[maxn + 5];
    bool vis[maxn + 5];
    unordered_map<int, LL> sum_miu;
    Sieve(int lim){//pre
    }
    LL get_h(int x){

    }
    LL get_g(int x){

    }
    LL get_s(LL x){
        if(x <= maxn) return miu[x];
        if(sum_miu.find(x) != sum_miu.end()) return sum_miu[x];
        LL asi = get_h(x);
        For(i, 2, x){
            LL tem = x / i, to = x / tem;
            asi -= (get_g(to) - get_g(i - 1)) * get_s(tem);
            i = to;
        }
        return sum_miu[x] = asi;
    }
} sieve(maxn);

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

LL n, as = 0;

int main(){
    // freopen("in.txt", "r", stdin);
    return 0;
}

```

```
}
```

3.14 辛普森积分

$$\int_r^l f(x)dx \approx \frac{r-l}{6}(f(l) + 4f(\frac{l+r}{2}) + f(r))$$

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-15
#define eeps 1e-9
using namespace std;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

DD get_val(DD x){

}

DD get_simpl(DD l, DD r){
    // return (get_val(l) * 989 + get_val((l * 7 + r) / 8.0) * 5888 -
    //         get_val((l * 3 + r) / 4.0) * 928 + get_val((l * 5 + r * 3) / 8.0)
    * 10496 -
    //         get_val((l + r) / 2.0) * 45440 + get_val(r) * 989 +
    //         get_val((r * 7 + l) / 8.0) * 5888 - get_val((r * 3 + l) / 4.0) *
    928 +
    //         get_val((r * 5 + l * 3) / 8.0) * 10496) /
    //         28350.0;
    // return (get_val(l) * 751 + get_val((l * 6 + r) / 7.0) * 3577 +
    //         get_val((l * 5 + r * 2) / 7.0) * 1323 + get_val((l + 4 + r * 3) /
    7.0) * 2989
    //         + get_val(r) * 751 + get_val((r * 6 + l) / 7.0) * 3577 +
    get_val((r * 5 + l * 2)
    //         / 7.0) * 1323 + get_val((r + 4 + l * 3) / 7.0) * 2989) /
```

```

//      17280.0;
// return (get_val(l) * 41 + get_val((5 * l + r) / 6.0) * 216 + get_val((2 *
l + r) / 3.0) * 27 +
//      get_val((l + r) / 2.0) * 272 + get_val((l + 2 * r) / 3.0) * 27 +
//      get_val((l + r * 5) / 6.0) * 216 + get_val(r) * 41) *
//      (r - l) / 840.0;
// return (get_val(l) + get_val(r) + get_val((l*2 + r) /
3.0)*3.0+get_val((l+r*2)/3.0)*3.0) * (r -
// l) / 8.0;
// return (get_val(l) + get_val(r) + 4 * get_val((l + r) / 2.0)) * (r - l) /
6.0;
}

DD simpl(DD l, DD r, DD now){
    DD mid = (l + r) / 2, teml = get_simpl(l, mid), temr = get_simpl(mid, r);
    if(fabs(now - teml - temr) <= eeps) return teml + temr;
    return simpl(l, mid, teml) + simpl(mid, r, temr);
}

int main(){
    // freopen("in.txt", "r", stdin);
    // cout << (DD)clock() / 1000 << "s" << endl;
    return 0;
}

```

3.15 Miller_Rabin & Pollard_Rho

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define i128 __int128_t
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
using namespace std;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

```

```

LL fp(LL x, LL y, LL P){
    LL res = 1;
    while(y){
        if(y & 1) res = (i128)res * x % P;
        x = (i128)x * x % P;
        y >>= 1;
    }
    return res;
}

LL gcd(LL x, LL y){return y ? gcd(y, x % y) : x;}

int Miller_Rabin(LL x){
    if(x < 3 || x % 2 == 0) return x == 2;
    LL u = x - 1, pri_bas[8] = {2, 325, 9375, 28178, 450775, 9780504,
1795265022};
    int t = 0;
    while(u % 2 == 0) u >>= 1, t++;
    For(o, 0, 6){
        LL now = fp(pri_bas[o], u, x);
        if(now <= 1) continue;
        For(i, 1, t){
            if(now == x - 1) break;
            if(i == t) return 0;
            now = (i128)now * now % x;
            if(now == 1) return 0;
        }
    }
    return 1;
}

LL Pollard_Rho(LL x){
    if(x == 4) return 2;
    if(Miller_Rabin(x)) return x;
    default_random_engine de;
    uniform_int_distribution<LL> rd(1, x - 1);
    while(666){
        LL c = rd(de);
        auto f = [&](LL _x){return ((i128)_x * _x + c) % x;};
        LL s = 0, t = 0;
        while(666){
            LL p = 1;
            For(i, 1, 128){
                s = f(s);
                t = f(f(t));
                if(s == t) break;
                LL tem = (i128)p * abs(s - t) % x;
                if(!tem) break;
                p = tem;
            }
            LL gc = gcd(p, x);
            if(gc > 1) return gc;
            if(s == t) break;
        }
    }
}

```

```

}

int main(){

}

```

3.16 五边形数，整数拆分

$$\prod_{k=1}^{\infty} (1 - x^k) = \sum_{k=-\infty}^{\infty} (-1)^k x^{\frac{3k^2-k}{2}}$$

4 数（树）据结构

4.1 树状数组

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 500000
using namespace std;

int n, m, tr[maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

void add(int x, int y){
    while(x <= n) tr[x] += y, x += x & -x;
}

int que(int x){
    int asi = 0;
    while(x) asi += tr[x], x -= x & -x;
}

```



```

        return asi;
    }

    int main(){
        //freopen("in", "r", stdin);
        read(n); read(m);
        For(i, 1, n){
            int x; read(x);
            add(i, x);
        }
        while(m--){
            int op, x, y;
            read(op); read(x); read(y);
            if(op == 1) add(x, y);
            else printf("%d\n", que(y) - que(x - 1));
        }
        return 0;
    }
}

```

4.2 线段树

以lpr为懒标记

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 100000
using namespace std;

int n, m, a[maxn + 5];
struct Tree{
    int tot;
    struct Node{
        int l, r, ls, rs;
        LL val, lpr;
    } poi[4 * maxn + 5];
    void init(){tot = 0;}
    void upd(int rt){poi[rt].val = poi[poi[rt].ls].val + poi[poi[rt].rs].val;}
    int build(int rt, int l, int r){
        poi[rt].lpr = 0;
        poi[rt].l = l;
        poi[rt].r = r;
        if(l == r){

```

```

        poi[rt].val = a[l];
        return rt;
    }
    int mid = l + r >> 1;
    poi[rt].ls = build(++tot, l, mid);
    poi[rt].rs = build(++tot, mid + 1, r);
    upd(rt);
    return rt;
}
void mson(int rt, LL val){
    poi[rt].val += (poi[rt].r - poi[rt].l + 1) * val;
    poi[rt].lpr += val;
}
void spr(int rt){
    mson(poi[rt].ls, poi[rt].lpr); mson(poi[rt].rs, poi[rt].lpr);
    poi[rt].lpr = 0;
}
void mdy(int rt, int l, int r, int val){
    if(l <= poi[rt].l && r >= poi[rt].r) return mson(rt, val);
    spr(rt);
    int mid = poi[rt].l + poi[rt].r >> 1;
    if(l <= mid) mdy(poi[rt].ls, l, r, val);
    if(r > mid) mdy(poi[rt].rs, l, r, val);
    upd(rt);
}
LL que(int rt, int l, int r){
    if(l <= poi[rt].l && r >= poi[rt].r) return poi[rt].val;
    spr(rt);
    int mid = poi[rt].l + poi[rt].r >> 1;
    LL asi = 0;
    if(l <= mid) asi += que(poi[rt].ls, l, r);
    if(r > mid) asi += que(poi[rt].rs, l, r);
    return asi;
}
} tr;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int main(){
    //freopen("in", "r", stdin);
    read(n); read(m);
    tr.init();
    For(i, 1, n) read(a[i]);
    tr.build(++tr.tot, 1, n);
    while(m--){
        int op, x, y, k;
        read(op);
        if(op == 1){

```

```

        read(x); read(y); read(k);
        tr.mdy(1, x, y, k);
    }
    else{
        read(x); read(y);
        printf("%11d\n", tr.que(1, x, y));
    }
}
return 0;
}

```

4.3 并查集

[P3367 【模板】并查集](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 10000
using namespace std;

int n, m, fa[maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int find(int x){
    return fa[x] == x ? x : fa[x] = find(fa[x]);
}

int main(){
    //freopen("in", "r", stdin);
    read(n); read(m);
    For(i, 1, n) fa[i] = i;
    while(m--){
        int op, x, y; read(op); read(x); read(y);
    }
}

```

```

        if(op == 1) fa[find(x)] = find(y);
        else{
            if(find(x) == find(y)) puts("Y");
            else puts("N");
        }
    }
    return 0;
}

```

4.4 ST表

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 100000
using namespace std;

int n, m, mx[2][maxn + 5][30];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int que(int l, int r){
    int tem = 0;
    while((1 << (tem + 1)) < r - l + 1) tem++;
    return max(mx[0][l][tem], mx[1][r][tem]);
}

int main(){
    //freopen("in", "r", stdin);
    read(n); read(m);
    For(i, 1, n) read(mx[0][i][0]), mx[1][i][0] = mx[0][i][0];
    For(i, 1, 20){
        For(j, 1, n){

```

```

        int to = 1 << (i - 1);
        mx[0][j][i] = mx[0][j][i - 1];
        if(j + to <= n && mx[0][j][i] < mx[0][j + to][i - 1]) mx[0][j][i] =
mx[0][j + to][i - 1];
        mx[1][j][i] = mx[1][j][i - 1];
        if(j - to >= 1 && mx[1][j][i] < mx[1][j - to][i - 1]) mx[1][j][i] =
mx[1][j - to][i - 1];
    }
}
while(m--){
    int l, r; read(l); read(r);
    printf("%d\n", que(l, r));
}
return 0;
}

```

4.5 平衡树

[P3369 【模板】普通平衡树](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 100000
using namespace std;

int t;

struct Node{int val, ch[2], cnt[2], fa;};
class Splay_Tree{
public:
    int tot = 1;
    Node poi[maxn + 5];
    Splay_Tree(){poi[1].val = inf;};
    void upd(int rt){poi[rt].cnt[1] = poi[rt].cnt[0] +
poi[poi[rt].ch[0]].cnt[1] + poi[poi[rt].ch[1]].cnt[1];}
    void rot(int rt){
        int fa = poi[rt].fa, is_ls = poi[fa].ch[0] == rt;
        poi[poi[fa].fa].ch[poi[fa].fa].ch[1] == fa ? rt].fa =
poi[fa].fa;
        poi[poi[fa].ch[!is_ls] = poi[rt].ch[is_ls]].fa = fa;
    }
}

```

```

        poi[poi[rt].ch[is_ls] = fa].fa = rt;
        upd(fa); upd(rt);
    }
    void spy(int x){while(poi[x].fa != 1) rot(x);}
    int find(int x){
        int now = 1;
        while(poi[now].val != x){
            int to = x > poi[now].val;
            if(!poi[now].ch[to]) poi[poi[now].ch[to] = ++tot] = {x, {0, 0},
{0, 0}, now};
            now = poi[now].ch[to];
        }
        spy(now);
        return now;
    }
    void add(int x, int val){
        poi[1].cnt[1] += val;
        int now = find(x);
        poi[now].cnt[0] += val;
        poi[now].cnt[1] += val;
    }
    int find_rank(int x){return poi[poi[find(x)].ch[0]].cnt[1] + 1;}
    int find_index(int x){
        int now = 1;
        while(666){
            if(x > poi[poi[now].ch[0]].cnt[1]){
                x -= poi[poi[now].ch[0]].cnt[1];
                if(x > poi[now].cnt[0]){
                    x -= poi[now].cnt[0];
                    now = poi[now].ch[1];
                }
                else break;
            }
            else now = poi[now].ch[0];
        }
        spy(now);
        return now;
    }
    int find_pre(int x){return find_index(find_rank(x) - 1);}
    int find_suf(int x){
        int now = find(x);
        return find_index(poi[poi[now].ch[0]].cnt[1] + poi[now].cnt[0] + 1);
    }
} tr;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int main(){

```

```

// freopen("in.txt", "r", stdin);
read(t);
while(t--){
    int opt, x; read(opt); read(x);
    if(opt == 1) tr.add(x, 1);
    if(opt == 2) tr.add(x, -1);
    if(opt == 3) printf("%d\n", tr.find_rank(x));
    if(opt == 4) printf("%d\n", tr.poi[tr.find_index(x)].val);
    if(opt == 5) printf("%d\n", tr.poi[tr.find_pre(x)].val);
    if(opt == 6) printf("%d\n", tr.poi[tr.find_suf(x)].val);
}
return 0;
}

```

4.6 LCT

[P4219 \[BJOI2014\]大融合](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 100000
using namespace std;

int n, q;
struct Node{int ch[2], cnt[2], fa, lpr;};
class Lct_Tree{
public:
    int tot = 0;
    Node poi[maxn + 5];
    Lct_Tree(){}
    void init(int _tot){
        tot = _tot;
        For(i, 1, tot) poi[i].cnt[0] = poi[i].cnt[1] = 1;
    }
    void upd(int rt){poi[rt].cnt[1] = poi[rt].cnt[0] +
poi[poi[rt].ch[0]].cnt[1] + poi[poi[rt].ch[1]].cnt[1];}
    void mson(int rt){
        swap(poi[rt].ch[0], poi[rt].ch[1]);
        poi[rt].lpr ^= 1;
    }
    void spr(int rt){

```

```

        if(poi[rt].lpr){
            mson(poi[rt].ch[0]);
            mson(poi[rt].ch[1]);
            poi[rt].lpr = 0;
        }
    }
    void chk(){
        cout << "asd" << endl;
        For(i, 1, n) cout << i << " " << is_rt(i) << " " << poi[i].fa << " "
<< poi[i].cnt[0] << endl;
    }
    void rot(int rt){
        int fa = poi[rt].fa;
        spr(fa); spr(rt);
        int is_ls = poi[fa].ch[0] == rt;
        if(!is_rt(fa)) poi[poi[fa].fa].ch[poi[poi[fa].fa].ch[1] == fa] = rt;
        poi[rt].fa = poi[fa].fa;
        poi[poi[fa].ch[!is_ls] = poi[rt].ch[is_ls]].fa = fa;
        poi[poi[rt].ch[is_ls] = fa].fa = rt;
        upd(fa); upd(rt);
    }
    bool is_rt(int rt){return poi[poi[rt].fa].ch[0] != rt &&
poi[poi[rt].fa].ch[1] != rt;}
    void spy(int x){while(!is_rt(x)) rot(x); spr(x);}
    void acc(int x){
        int now = x, pre = 0;
        while(now){
            spy(now);
            poi[now].cnt[0] += poi[poi[now].ch[1]].cnt[1] - poi[pre].cnt[1];
            poi[now].ch[1] = pre;
            pre = now;
            now = poi[now].fa;
        }
        spy(x);
    }
    void mkrt(int x){acc(x); mson(x);}
    void link(int u, int v){
        mkrt(u); mkrt(v);
        poi[v].fa = u;
        poi[u].cnt[0] += poi[v].cnt[1];
        poi[u].cnt[1] += poi[v].cnt[1];
    }
    void cut(int u, int v){
        mkrt(u); acc(v);
        poi[v].cnt[1] -= poi[u].cnt[1];
        poi[v].ch[0] = poi[u].fa = 0;
    }
} tr;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());

```



```

        if(ok) x = -x;
    }

    int main(){
        // freopen("in.txt", "r", stdin);
        read(n); read(q);
        tr.init(n);
        while(q--){
            char c; cin >> c;
            int x, y; read(x); read(y);
            if(c == 'A') tr.link(x, y);
            else{
                tr.mkrt(x); tr.acc(y);
                printf("%11d\n", 111 * tr.poi[x].cnt[1] * (tr.poi[y].cnt[1] -
tr.poi[x].cnt[1]));
            }
        }
        return 0;
    }

```

4.7 bitset

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define U unsigned
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
using namespace std;

template <class T>
void read(T& x) {
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

struct Bit : vector<U LL> {
    Bit operator <=< (int _x) {

```

```

    int tem = _x / 64;
    Rof(i, size() - 1, tem) (*this)[i] = (*this)[i - tem];
    For(i, 0, min((int)size(), tem) - 1) (*this)[i] = 0;
    _x &= 63;
    if(_x) {
        Rof(i, size() - 1, 1) (*this)[i] = ((*this)[i] << _x) | ((*this)[i -
1] >> (64 - _x));
        (*this)[0] <<= _x;
    }
    return (*this);
}

Bit operator << (cst int &_x) {
    Bit tem = *this;
    tem <<= _x;
    return tem;
}

Bit operator >>= (int _x) {
    int tem = _x / 64;
    For(i, 0, size() - 1 - tem) (*this)[i] = (*this)[i + tem];
    For(i, max(0, (int)size() - tem), size() - 1) (*this)[i] = 0;
    _x &= 63;
    if(_x) {
        For(i, 0, size() - 2) (*this)[i] = ((*this)[i] >> _x) | (((*this)[i +
1] & ((1ull << _x) - 1)) << (64 - _x));
        (*this)[size() - 1] >>= _x;
    }
    return (*this);
}

Bit operator >> (cst int &_x) {
    Bit tem = *this;
    tem >>= _x;
    return tem;
}

Bit operator &= (cst Bit &_x) { For(i, 0, size() - 1) (*this)[i] &= _x[i];
return (*this); }

Bit operator |= (cst Bit &_x) { For(i, 0, size() - 1) (*this)[i] |= _x[i];
return (*this); }

Bit operator ^= (cst Bit &_x) { For(i, 0, size() - 1) (*this)[i] ^= _x[i];
return (*this); }

friend Bit operator & (Bit _x, cst Bit &_y) { return _x &= _y; }
friend Bit operator | (Bit _x, cst Bit &_y) { return _x |= _y; }
friend Bit operator ^ (Bit _x, cst Bit &_y) { return _x ^= _y; }
int get_bit(int _x) { return ((*this)[_x >> 6] >> (_x & 63)) & 1; }
void init(cst int &_x) {
    resize((( _x - 1) >> 6) + 1);
    For(i, 0, size() - 1) (*this)[i] = 0;
}

};

int main() {
    //freopen("in", "r", stdin);
}

```

5 图论 (吐子)

5.1 LCA (最近公共祖先)

[P3379 【模板】最近公共祖先 \(LCA\)](#)

采用倍增:

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 1000000
using namespace std;

int n, m, s;
struct Tree{
    int len, hd[maxn + 5];
    struct Edge{
        int v, net;
    }e[maxn + 5];
    void init(){
        memset(hd, -1, sizeof hd);
        len = 0;
    }
    void add(int u, int v){
        e[len] = (Edge){v, hd[u]};
        hd[u] = len++;
    }
    int dep[maxn + 5], to[maxn + 5][30];
    void dfs(int u, int fa){
        dep[u] = dep[fa] + 1;
        to[u][0] = fa;
        For(i, 1, 20) to[u][i] = to[to[u][i - 1]][i - 1];
        Tra(u, i){
            int v = e[i].v;
            if(v == fa) continue;
            dfs(v, u);
        }
    }
    int lca(int x, int y){
        if(dep[x] < dep[y]) swap(x, y);
```

```

        Rof(i, 20, 0) if(dep[to[x][i]] >= dep[y]){
            x = to[x][i];
            if(dep[x] == dep[y]) break;
        }
        if(x == y) return x;
        Rof(i, 20, 0) if(to[x][i] != to[y][i]){
            x = to[x][i];
            y = to[y][i];
        }
        return to[x][0];
    }
} tr;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int main(){
    //freopen("in", "r", stdin);
    read(n); read(m); read(s);
    tr.init();
    For(i, 1, n - 1){
        int u, v; read(u); read(v);
        tr.add(u, v); tr.add(v, u);
    }
    tr.dfs(s, 0);
    while(m--){
        int x, y; read(x); read(y);
        printf("%d\n", tr.lca(x, y));
    }
    return 0;
}

```

5.2 最小生成树

[P3366 【模板】最小生成树](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first

```

```

#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 5000
#define maxm 200000
using namespace std;

int n, m, fa[maxn + 5], cnt, as = 0;
struct Edge{
    int u, v, w;
}e[maxm + 5];
bool operator < (Edge &x, Edge &y){return x.w < y.w;}

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

int find(int x){return fa[x] == x ? x : fa[x] = find(fa[x]);}

int main(){
    //freopen("in", "r", stdin);
    read(n); read(m); cnt = n;
    For(i, 1, n) fa[i] = i;
    For(i, 1, m){
        int u, v, w; read(u); read(v); read(w);
        e[i] = (Edge){u, v, w};
    }
    sort(e + 1, e + m + 1);
    For(i, 1, m){
        int u = e[i].u, v = e[i].v, w = e[i].w;
        if(find(u) == find(v)) continue;
        fa[find(u)] = find(v);
        cnt--;
        as += w;
    }
    if(cnt != 1) puts("orz");
    else printf("%d\n", as);
    return 0;
}

```

5.3 最短路

采用Dijkstra

[P4779 【模板】单源最短路径（标准版）](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)

```

```

#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 500000
using namespace std;

int n, m, s, hd[maxn + 5], len = 0;
struct Edge{
    int v, w, net;
}e[maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

void add(int u, int v, int w){
    e[len] = (Edge){v, w, hd[u]};
    hd[u] = len++;
}

int vis[maxn + 5];
LL dis[maxn + 5];
void dij(){
    memset(dis, inf, sizeof dis);
    priority_queue<pair<int, int> > q;
    dis[s] = 0;
    q.push(mp(0, s));
    while(q.size()){
        int u = q.top().sec; q.pop();
        if(vis[u]) continue;
        vis[u] = 1;
        Tra(u, i){
            int v = e[i].v, w = e[i].w;
            if(dis[u] + w < dis[v]){
                dis[v] = dis[u] + w;
                q.push(mp(-dis[v], v));
            }
        }
    }
}

```

```

int main(){
    //freopen("in", "r", stdin);
    memset(hd, -1, sizeof hd);
    read(n); read(m); read(s);
    For(i, 1, m){
        int u, v, w; read(u); read(v); read(w);
        add(u, v, w);
    }
    dij();
    For(i, 1, n) printf("%lld ", dis[i]);
    return 0;
}

```

5.4 树链剖分

[P2590 \[ZJOI2008\]树的统计](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 30000
using namespace std;

int n, hd[maxn + 5], len = 0, w[maxn + 5], q;
int mxs[maxn + 5], sz[maxn + 5], fa[maxn + 5], id[maxn + 5], ffa[maxn + 5], cnt = 0, to[maxn + 5], dep[maxn + 5];
struct Edge{
    int v, net;
}e[2 * maxn + 5];
struct Tree{
    int tot;
    struct Node{
        int l, r, ls, rs, mx, sum;
    } poi[4 * maxn + 5];
    void init(){
        tot = 0;
    }
    void upd(int rt){
        poi[rt].sum = poi[poi[rt].ls].sum + poi[poi[rt].rs].sum;
        poi[rt].mx = max(poi[poi[rt].ls].mx, poi[poi[rt].rs].mx);
    }
}

```

```

}
int build(int rt, int l, int r){
    poi[rt].l = l; poi[rt].r = r;
    if(l == r){
        poi[rt].mx = poi[rt].sum = w[to[l]];
        return rt;
    }
    int mid = l + r >> 1;
    poi[rt].ls = build(++tot, l, mid);
    poi[rt].rs = build(++tot, mid + 1, r);
    upd(rt);
    return rt;
}
void mdy(int rt, int x, int val){
    if(poi[rt].l == poi[rt].r){
        poi[rt].mx = poi[rt].sum = val;
        return;
    }
    int mid = poi[rt].l + poi[rt].r >> 1;
    if(x <= mid) mdy(poi[rt].ls, x, val);
    else mdy(poi[rt].rs, x, val);
    upd(rt);
}
int que(int rt, int l, int r){
    if(l <= poi[rt].l && r >= poi[rt].r) return poi[rt].mx;
    int mid = poi[rt].l + poi[rt].r >> 1, asi = -inf;
    if(l <= mid) asi = max(asi, que(poi[rt].ls, l, r));
    if(r > mid) asi = max(asi, que(poi[rt].rs, l, r));
    return asi;
}
int que1(int rt, int l, int r){
    if(l <= poi[rt].l && r >= poi[rt].r) return poi[rt].sum;
    int mid = poi[rt].l + poi[rt].r >> 1, asi = 0;
    if(l <= mid) asi += que1(poi[rt].ls, l, r);
    if(r > mid) asi += que1(poi[rt].rs, l, r);
    return asi;
}
} tr;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

void add(int u, int v){
    e[len] = (Edge){v, hd[u]};
    hd[u] = len++;
}

void dfs(int u, int pre){
    sz[u] = 1;

```



```

dep[u] = dep[pre] + 1;
fa[u] = pre;
Tra(u, i){
    int v = e[i].v;
    if(v == pre) continue;
    dfs(v, u);
    sz[u] += sz[v];
    if(sz[v] > sz[mxs[u]]) mxs[u] = v;
}
}

void dfs1(int u, int pre, int top){
    to[id[u] = ++cnt] = u;
    ffa[u] = top;
    if(mxs[u]) dfs1(mxs[u], u, top);
    Tra(u, i){
        int v = e[i].v;
        if(v == pre || v == mxs[u]) continue;
        dfs1(v, u, v);
    }
}

int que(int u, int v){
    int asi = -inf;
    while(ffa[u] != ffa[v]){
        if(dep[ffa[u]] < dep[ffa[v]]) swap(u, v);
        asi = max(asi, tr.que(1, id[ffa[u]], id[u]));
        u = fa[ffa[u]];
    }
    if(dep[u] < dep[v]) swap(u, v);
    return max(asi, tr.que(1, id[v], id[u]));
}

int que1(int u, int v){
    int asi = 0;
    while(ffa[u] != ffa[v]){
        if(dep[ffa[u]] < dep[ffa[v]]) swap(u, v);
        asi += tr.que1(1, id[ffa[u]], id[u]);
        u = fa[ffa[u]];
    }
    if(dep[u] < dep[v]) swap(u, v);
    return asi + tr.que1(1, id[v], id[u]);
}

int main(){
    //freopen("in", "r", stdin);
    memset(hd, -1, sizeof hd);
    read(n);
    For(i, 1, n - 1){
        int u, v; read(u); read(v);
        add(u, v); add(v, u);
    }
    For(i, 1, n) read(w[i]);
    dfs(1, 0); dfs1(1, 0, 1);
    tr.init();
    tr.build(++tr.tot, 1, n);
    read(q);
}

```

```

while(q--){
    char s[15];
    int x, y;
    scanf("%s", s + 1);
    read(x); read(y);
    if(s[1] == 'C') tr.mdy(1, id[x], y);
    if(s[2] == 'M') printf("%d\n", que(x, y));
    if(s[2] == 'S') printf("%d\n", que1(x, y));
}
return 0;
}

```

5.5 网络最大流

[P3376 【模板】网络最大流](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 5000
using namespace std;

int n, m, s, t;
struct Edge { int v, w, bas, net; };
class Graph {
public:
    int hd[maxn + 5], hd_bac[maxn + 5], len = 0, dep[maxn + 5];
    Edge e[2 * maxn + 5];
    Graph() { memset(hd, -1, sizeof hd); }
    void back() { memcpy(hd_bac, hd, sizeof hd); }
    void reco() { memcpy(hd, hd_bac, sizeof hd); }
    void addi(int u, int v, int w) {
        e[len] = { v, w, 0, hd[u] };
        hd[u] = len++;
    }
    void add(int u, int v, int w) {
        addi(u, v, w);
        addi(v, u, 0);
    }
} G;

template <class T>
void read(T &x) {

```

```

char ch;
bool ok;
for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
if(ok) x = -x;
}

bool bfs() {
    G.reco();
    queue<int> q;
    memset(G.dep, 0, sizeof G.dep);
    G.dep[s] = 1;
    q.push(s);
    while(q.size()) {
        int u = q.front(); q.pop();
        for(int i = G.hd[u]; ~i; i = G.e[i].net) {
            int v = G.e[i].v, w = G.e[i].w - G.e[i].bas;
            if(!w || G.dep[v]) continue;
            G.dep[v] = G.dep[u] + 1;
            if(v == t) return 1;
            q.push(v);
        }
    }
    return 0;
}

int dfs(int u, int flow) {
    // cout << flow << endl;
    if(u == t) return flow;
    int asi = 0;
    for(int &i = G.hd[u]; ~i; i = G.e[i].net) {
        int v = G.e[i].v, w = G.e[i].w - G.e[i].bas;
        if(!w || G.dep[v] != G.dep[u] + 1) continue;
        int tem = dfs(v, min(w, flow));
        if(!tem) continue;
        asi += tem;
        flow -= tem;
        G.e[i].bas += tem;
        G.e[i ^ 1].bas -= tem;
        if(!flow) break;
    }
    return asi;
}

LL dinic() {
    LL asi = 0;
    // bfs(); cout << dfs(s, inf) << endl;
    // bfs(); cout << dfs(s, inf) << endl;
    G.back();
    while(bfs()) asi += dfs(s, inf);
    return asi;
}

int main() {
    // freopen("in.txt", "r", stdin);

```

```

read(n); read(m); read(s); read(t);
For(i, 1, m) {
    int u, v, w; read(u); read(v); read(w);
    G.add(u, v, w); G.add(v, u, 0);
}
printf("%lld\n", dinic());
return 0;
}

```

5.6 最小费用最大流

[P3381 【模板】最小费用最大流](#)

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 50000
using namespace std;

int n, m, s, t;
struct Edge{int v, w, c, bas, net;};
class Graph{
public:
    int hd[maxn + 5], len = 0, flow[maxn + 5], pre[maxn + 5];
    LL cos[maxn + 5];
    Edge e[2 * maxn + 5];
    Graph(){memset(hd, -1, sizeof hd);}
    void addi(int u, int v, int w, int c) {
        e[len] = { v, w, c, 0, hd[u] };
        hd[u] = len++;
    }
    void add(int u, int v, int w, int c) {
        addi(u, v, w, c);
        addi(v, u, 0, -c);
    }
} G;

template <class T>
void read(T &x){
    char ch;

```

```

bool ok;
for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok =
1;
for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
if(ok) x = -x;
}

int inq[maxn + 5];
int bfs(){
    queue<int> q;
    memset(G.cos, inf, sizeof G.cos);
    G.cos[s] = 0;
    G.flow[s] = inf;
    q.push(s); inq[s] = 1;
    while(q.size()){
        int u = q.front(); q.pop(); inq[u] = 0;
        for(int i = G.hd[u]; ~i; i = G.e[i].net){
            int v = G.e[i].v, w = G.e[i].w - G.e[i].bas, c = G.e[i].c;
            if(!w || G.cos[v] <= G.cos[u] + c) continue;
            G.cos[v] = G.cos[u] + c;
            G.flow[v] = min(w, G.flow[u]);
            G.pre[v] = i;
            if(!inq[v]) q.push(v), inq[v] = 1;
        }
    }
    if(G.cos[t] == Inf) return 0;
    return G.flow[t];
}

pair<LL, LL> ek(){
    pair<LL, LL> asi = {0, 0};
    int flow = 0;
    int tot = 0;
    while(flow = bfs()){
        tot++;
        asi.fir += flow;
        asi.sec += 1ll * flow * G.cos[t];
        int now = t;
        while(now != s){
            G.e[G.pre[now]].bas += flow;
            G.e[G.pre[now] ^ 1].bas -= flow;
            now = G.e[G.pre[now] ^ 1].v;
        }
    }
    return asi;
}

int main(){
    // freopen("in.txt", "r", stdin);
    read(n); read(m); read(s); read(t);
    For(i, 1, m){
        int u, v, w, c; read(u); read(v); read(w); read(c);
        G.add(u, v, w, c); G.add(v, u, 0, -c);
    }
    pair<LL, LL> asi = ek();
    printf("%lld %lld\n", asi.fir, asi.sec);
}

```

```
return 0;
}
```

上下界网络流

概述

上下界网络流本质是给流量网络的每一条边设置了流量上界 $c(u, v)$ 和流量下界 $b(u, v)$ 。也就是说，一种可行的流必须满足 $b(u, v) \leq f(u, v) \leq c(u, v)$ 。同时必须满足除了源点和汇点之外的其余点流量平衡。

根据题目要求，我们可以使用上下界网络流解决不同问题。

无源汇上下界可行流

给定无源汇流量网络 G 。询问是否存在一种标定每条边流量的方式，使得每条边流量满足上下界同时每一个点流量平衡。

不妨假设每条边已经流了 $b(u, v)$ 的流量，设其为初始流。同时我们在新图中加入 u 连向 v 的流量为 $c(u, v) - b(u, v)$ 的边。考虑在新图上进行调整。

由于最大流需要满足初始流量平衡条件（最大流可以看成是下界为 0 的上下界最大流），但是构造出来的初始流很有可能不满足初始流量平衡。假设一个点初始流入流量减初始流出流量为 M 。

若 $M = 0$ ，此时流量平衡，不需要附加边。

若 $M > 0$ ，此时入流量过大，需要新建附加源点 S' ， S' 向其连流量为 M 的附加边。

若 $M < 0$ ，此时出流量过大，需要新建附加汇点 T' ，其向 T' 连流量为 $-M$ 的附加边。

如果附加边满流，说明这一个点的流量平衡条件可以满足，否则这个点的流量平衡条件不满足。（因为原图加上附加流之后才会满足原图中的流量平衡。）

在建图完毕之后跑 S' 到 T' 的最大流，若 S' 连出去的边全部满流，则存在可行流，否则不存在。

有源汇上下界可行流

给定有源汇流量网络 G 。询问是否存在一种标定每条边流量的方式，使得每条边流量满足上下界同时除了源点和汇点每一个点流量平衡。

假设源点为 S ，汇点为 T 。

则我们可以加入一条 T 到 S 的上界为 ∞ ，下界为 0 的边转化为无源汇上下界可行流问题。

若有解，则 S 到 T 的可行流流量等于 T 到 S 的附加边的流量。

有源汇上下界最大流

给定有源汇流量网络 G 。询问是否存在一种标定每条边流量的方式，使得每条边流量满足上下界同时除了源点和汇点每一个点流量平衡。如果存在，询问满足标定的最大流量。

我们找到网络上的任意一个可行流。如果找不到解就可以直接结束。

否则我们考虑删去所有附加边之后的残量网络并且在网络上进行调整。

我们在残量网络上再跑一次 S 到 T 的最大流，将可行流流量和最大流流量相加即为答案。

S 到 T 的最大流直接在跑完有源汇上下界可行的残量网络上跑。

千万不可以在原来的流量网络上跑。

有源汇上下界最小流

给定有源汇流量网络 G 。询问是否存在一种标定每条边流量的方式，使得每条边流量满足上下界同时除了源点和汇点每一个点流量平衡。如果存在，询问满足标定的最小流量。

类似的，我们考虑将残量网络中不需要的流退掉。

我们找到网络上的任意一个可行流。如果找不到解就可以直接结束。

否则我们考虑删去所有附加边之后的残量网络。

我们在残量网络上再跑一次 T 到 S 的最大流，将可行流流量减去最大流流量即为答案。

5.7 二分图最大权完美匹配

[P6577 【模板】二分图最大权完美匹配](#)

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 500
using namespace std;

int n, m;
LL as = 0, edge[maxn + 5][maxn + 5];

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

LL lv[maxn + 5], rv[maxn + 5], sla[maxn + 5], sla_to[maxn + 5];
int lto[maxn + 5], rto[maxn + 5], okl[maxn + 5], okr[maxn + 5];
void sol(int x){
    memset(okl, 0, sizeof okl);
    memset(okr, 0, sizeof okr);
    okl[x] = 1;
```

```

For(i, 1, n) sla[i] = 111 * lv[x] + rv[i] - edge[x][i], sla_to[i] = x;
while(666){
    int mn = 0;
    For(i, 1, n) if(!okr[i] && (!mn || sla[i] < sla[mn])) mn = i;
    LL tem = sla[mn];
    For(i, 1, n){
        if(okl[i]) lv[i] -= tem;
        if(okr[i]) rv[i] += tem;
        else sla[i] -= tem;
    }
    if(!rto[mn]){
        int now = sla_to[mn], pre = mn;
        while(now){
            int ppre = lto[now], nnow = sla_to[ppre];
            lto[now] = pre;
            rto[pre] = now;
            now = nnow;
            pre = ppre;
        }
        break;
    }
    else{
        okl[rto[mn]] = 1;
        okr[mn] = 1;
        For(i, 1, n) if(!okr[i] && sla[i] > 111 * lv[rto[mn]] + rv[i] -
edge[rto[mn]][i]){
            sla[i] = 111 * lv[rto[mn]] + rv[i] - edge[rto[mn]][i];
            sla_to[i] = rto[mn];
        }
    }
}

}

int main(){
    // freopen("in.txt", "r", stdin);
    read(n); read(m);
    For(i, 1, n) For(j, 1, n) edge[i][j] = -Inf;
    For(i, 1, m){
        int u, v, w; read(u); read(v); read(w);
        edge[u][v] = w;
    }
    memset(lv, inf, sizeof lv);
    memset(rv, 0, sizeof rv);
    For(i, 1, n) sol(i);
    For(i, 1, n) as += edge[rto[i]][i];
    printf("%lld\n", as);
    For(i, 1, n) printf("%d ", rto[i]);
    return 0;
}

```


5.8 一般图最大匹配（带花树）

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
#define maxn 600
using namespace std;

int t, n, m;

template <class T>
void read(T &x){
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

vector<int> ch[maxn + 5];
int col[maxn + 5], pre[maxn + 5], to[maxn + 5];
void sol(int u){
    if(to[pre[u]]) sol(to[pre[u]]);
    to[u] = pre[u];
    to[pre[u]] = u;
}

int vis[maxn + 5], tot = 0, id[maxn + 5];
int lca(int u, int v){
    tot++;
    while(666){
        if(v){
            if(vis[v] == tot) return v;
            vis[v] = tot;
            v = id[pre[to[v]]];
        }
        swap(u, v);
    }
}

queue<int> q;
void blossom(int u, int v, int lc){
    while(id[v] != lc){
        pre[v] = u;
```

```

        u = to[v];
        if(col[u] == 2){
            col[u] = 1;
            q.push(u);
        }
        id[u] = id[v] = lc;
        v = pre[u];
    }
}

bool bfs(int rt){
    memset(pre, 0, sizeof pre);
    memset(col, 0, sizeof col);
    For(i, 1, 2 * n + m) id[i] = i;
    while(q.size()) q.pop();
    q.push(rt);
    col[rt] = 1;
    while(q.size()){
        int u = q.front(); q.pop();
        for(auto v : ch[u]){
            if(!col[v]){
                pre[v] = u;
                col[v] = 2;
                if(!to[v]){
                    sol(v);
                    return 1;
                }
                col[to[v]] = 1;
                q.push(to[v]);
            }
            else if(col[v] == 1 && id[u] != id[v]){
                int lc = lca(id[u], id[v]);
                blossom(u, v, lc);
                blossom(v, u, lc);
            }
        }
    }
    return 0;
}

int main(){
    // freopen("in.txt", "r", stdin);
    return 0;
}

```

5.9 tarjan

无向图求点双

```

#define maxn 2000000
vector<pair<int, int> > ch[maxn + 5];
int cut[maxn + 5];
int dfn[maxn + 5], low[maxn + 5], tim = 0;
int st[maxn + 5], top = 0;
vector<vector<int> > as;

```

```

void tarjan(int u, int pre) {
    dfn[u] = low[u] = ++tim;
    st[++top] = u;
    int sz = 0;
    for(auto [v, id] : ch[u]) if(id != pre) {
        if(!dfn[v]) {
            sz++;
            tarjan(v, id);
            low[u] = min(low[u], low[v]);
            if(low[v] >= dfn[u]) {
                cut[u] = 1;
                vector<int> vec;
                while(st[top + 1] != v) vec.pb(st[top--]);
                vec.pb(u);
                as.pb(vec);
            }
        }
        else low[u] = min(low[u], dfn[v]);
    }
    if(!pre && !sz) as.pb({ u });
}

void mian() {
    read(n); read(m);
    For(i, 1, m) {
        int u, v; read(u); read(v);
        ch[u].pb({ v, i });
        ch[v].pb({ u, i });
    }
    For(i, 1, n) if(!dfn[i]) tarjan(i, 0);
    printf("%d\n", as.size());
    for(auto v : as) {
        printf("%d ", v.size());
        for(auto i : v) printf("%d ", i); puts("");
    }
}

```

无向图求边双

```

#define maxn 2000000
vector<pair<int, int> > ch[maxn + 5];
int cut[maxn + 5];
int dfn[maxn + 5], low[maxn + 5], tim = 0;
int st[maxn + 5], top = 0;
vector<vector<int> > as;
void tarjan(int u, int pre) {
    dfn[u] = low[u] = ++tim;
    st[++top] = u;
    int sz = 0;
    for(auto [v, id] : ch[u]) if(id != pre) {
        if(!dfn[v]) {
            sz++;
            tarjan(v, id);
            low[u] = min(low[u], low[v]);

```

```

        if(low[v] == dfn[v]) {
            cut[id] = 1;
            vector<int> vec;
            while(st[top + 1] != v) vec.pb(st[top--]);
            as.pb(vec);
        }
    }
    else low[u] = min(low[u], dfn[v]);
}
if(!pre) {
    vector<int> vec;
    while(top) vec.pb(st[top--]);
    as.pb(vec);
}
}
void mian() {
    read(n); read(m);
    For(i, 1, m) {
        int u, v; read(u); read(v);
        ch[u].pb({ v, i });
        ch[v].pb({ u, i });
    }
    For(i, 1, n) if(!dfn[i]) tarjan(i, 0);
    printf("%d\n", as.size());
    for(auto v : as) {
        printf("%d ", v.size());
        for(auto i : v) printf("%d ", i); puts("");
    }
}
}

```

有向图求强连通分量

```

void dfs(Graph &gra, int u){
    dfn[u] = low[u] = ++tim;
    inst[st[++top] = u] = 1;
    gTra(gra, u, i){
        int v = gra.e[i].v;
        if(!dfn[v]){
            dfs(gra, v);
            low[u] = min(low[u], low[v]);
        }
        else if(inst[v]) low[u] = min(low[u], low[v]);
    }
    if(low[u] == dfn[u]){
        tot_col++;
        while(st[top] != u){
            int tem = st[top--];
            col[tem] = tot_col;
            inst[tem] = 0;
        }
        col[st[top--]] = tot_col;
        inst[u] = 0;
    }
}

```

```
}
```

```
#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f
#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
using namespace std;

template <class T>
void read(T& x) {
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

#define maxn 10000
int T, n, m;
vector<int> ch[3][maxn + 5];

int st[maxn + 5], tp = 0, vis[maxn + 5], ok[maxn + 5];
void dfs(int u) {
    vis[u] = 1;
    ok[u] = 1;
    for(auto v : ch[0][u]) if(!vis[v]) dfs(v);
    st[++tp] = u;
}

vector<int> poi[maxn + 5];
int col[maxn + 5], d[maxn + 5];
void dfs1(int u, int c) {
    poi[c].pb(u);
    col[u] = c;
    ok[u] = 0;
    for(auto v : ch[1][u]) if(ok[v]) dfs1(v, c);
}

void mian() {
    read(n); read(m);
    For(i, 1, m) {
        int u, v; read(u); read(v);
```

```

        ch[0][u].pb(v);
        ch[1][v].pb(u);
    }
    tp = 0;
    int cnt = 0;
    For(i, 1, n) if(!vis[i]) {
        tp = 0;
        dfs(i);
        Rof(j, tp, 1) if(ok[st[j]]) dfs1(st[j], ++cnt);
    }
    // For(i, 1, n) cout << col[i] << " "; cout << endl;
    // For(u, 1, n) for(auto v : ch[0][u]) if(col[u] != col[v]) ch[2]
[col[u]].pb(col[v]), d[col[v]]++;
    For(i, 1, cnt) sort(poi[i].begin(), poi[i].end());
    printf("%d\n", cnt);
    For(i, 1, n) if(vis[i]) {
        for(auto v : poi[col[i]]) printf("%d ", v), vis[v] = 0;
        puts("");
    }
}

int main() {
    // freopen("in.txt", "r", stdin);
    // read(T);
    T = 1;
    while(T--) {
        mian();
        For(i, 1, n) {
            d[i] = 0;
            vis[i] = 0;
            For(o, 0, 2) ch[o][i].resize(0);
        }
    }
}

```

6 计算几何

6.1 半平面交 && 凸包

```

#include<bits/stdc++.h>
#define For(i, a, b) for(int i = (a), en = (b); i <= en; ++i)
#define Rof(i, a, b) for(int i = (a), en = (b); i >= en; --i)
#define Tra(u, i) for(int i = hd[u]; ~i; i = e[i].net)
#define cst const
#define LL long long
#define DD double
#define LD long double
#define pb push_back
#define mp make_pair
#define fir first
#define sec second
#define inf 0x3f3f3f3f

```

```

#define Inf 0x3f3f3f3f3f3f3f3f
#define eps 1e-12
using namespace std;

template <class T>
void read(T& x) {
    char ch;
    bool ok;
    for(ok = 0, ch = getchar(); !isdigit(ch); ch = getchar()) if(ch == '-') ok = 1;
    for(x = 0; isdigit(ch); x = x * 10 + ch - '0', ch = getchar());
    if(ok) x = -x;
}

#define maxn 100000
namespace Geometry {

    int dcmp(DD x) {
        if(fabs(x) <= eps) return 0;
        return x < 0 ? -1 : 1;
    }

    struct Vector {
        DD x, y;
        DD len() { return sqrt(x * x + y * y); }
        DD len2() { return x * x + y * y; }
        friend Vector operator + (Vector _x, cst Vector &_y) { return { _x.x + _y.x, _x.y + _y.y }; }
        friend Vector operator - (Vector _x, cst Vector &_y) { return { _x.x - _y.x, _x.y - _y.y }; }
        friend Vector operator * (cst DD &_x, cst Vector &_y) { return { _x * _y.x, _x * _y.y }; }
        friend DD operator * (cst Vector &_x, cst Vector &_y) { return _x.x * _y.x + _x.y * _y.y; }
        friend DD operator ^ (cst Vector &_x, cst Vector &_y) { return _x.x * _y.y - _x.y * _y.x; }
        Vector operator ~ () cst { return { -y, x }; }
    };

    struct Line {
        Vector x, v;
        Line() {}
        Line(cst Vector &_x, cst Vector &_y) { x = _x; v = _y - _x; }
        friend Vector operator & (cst Line &_x, cst Line &_y) {
            DD t = (_y.x - _x.x) * (~_y.v) / (_x.v * ~_y.v + eps);
            return _x.x + t * _x.v;
        }
    };

    DD get_s(Vector *a, int n) {
        DD s = a[n] ^ a[1];
        For(i, 1, n - 1) s += a[i] ^ a[i + 1];
        return fabs(s) / 2;
    }

    namespace HalfPlaneCross {
        int n;
    }
}

```

```

Line a[maxn + 5];
int dq[maxn + 5], l = 1, r = 0;
Vector cro[maxn + 5];
DD get_cro_s() {
    For(i, 1, r - 1) cro[i] = a[dq[l + i - 1]] & a[dq[l + i]];
    cro[r - 1 + 1] = a[dq[r]] & a[dq[l]];
    return get_s(cro, r - 1 + 1);
}
void mian() {
    For(i, 1, n) a[i].v.y += i * eps;
    sort(a + 1, a + n + 1, [](cst Line &x, cst Line &y) {return
atan2(_x.v.y, _x.v.x) < atan2(_y.v.y, _y.v.x)});
    For(i, 1, n) {
        while(l < r && dcmp(a[i].v ^ ((a[dq[r]] & a[dq[r - 1]]) -
a[i].x)) <= 0) r--;
        dq[++r] = i;
    }
    while(l < r && dcmp(a[dq[l]].v ^ ((a[dq[r]] & a[dq[r - 1]]) -
a[dq[l]].x)) <= 0) r--;
    while(l < r && dcmp(a[dq[l + 1]].v ^ ((a[dq[l]] & a[dq[r]]) - a[dq[l
+ 1]].x)) <= 0) l++;
}
void init(Line *_a, int _n) {
    l = 1; r = 0;
    n = _n;
    For(i, 1, n) a[i] = *_a[i];
    mian();
}
}

namespace Convex {
    int n, l, r;
    Vector a[maxn + 5];
    int dq[maxn + 5];
    DD get_len() {
        DD res = (a[dq[l]] - a[dq[r]]).len();
        For(i, 1, r - l) res += (a[dq[l + i - 1]] - a[dq[l + i]]).len();
        return res;
    }
    void mian() {
        sort(a + 1, a + n + 1, [](cst Vector &x, cst Vector &y) {return
_x.x == _y.x ? _x.y < _y.y : _x.x < _y.x});
        For(i, 1, n) {
            while(l < r && dcmp((a[dq[r]] - a[dq[r - 1]]) ^ (a[i] -
a[dq[r]])) <= 0) r--;
            dq[++r] = i;
        }
        int tem = r;
        Rof(i, n, 1) {
            while(r > tem && dcmp((a[dq[r]] - a[dq[r - 1]]) ^ (a[i] -
a[dq[r]])) <= 0) r--;
            dq[++r] = i;
        }
    }
}
void init(Vector *_a, int _n) {
    l = 1, r = 0;

```



```
        n = _n;
        For(i, 1, n) a[i] = _a[i];
    }
}

int main() {
    // freopen("in.txt", "r", stdin);
    return 0;
}
```