万能初始模板（读入优化）

```
/**********************
*    Problem:      *
*    User: mzg1816_LRL *
*    Language: C++     *
*    Date: 2019.   *
*    算法: *
**********************/
#define ri register int
#define ll long long
#define ci const int&
#define rep(i,a,b) for(register int i=(a);i<=(b);++i)
#define dwn(i,a,b) for(register int i=(a);i>=(b);--i)
#define ee(i,a) for(register int i=head[a];i;i=e[i].nxt)
#include<bits/stdc++.h>
using namespace std;
const int N=1055;
const int M=2055;
const int inf=0x3f3f3f3f;
char ss[1<<21],*A=ss,*B=ss,c;//int eof=1;
inline char gc(){return A==B&&(B=(A=ss)+fread(ss,1,1<<21,stdin),A==B)?EOF:*A++;}
struct RD{
    template<class T>
     RD&operator>>(T&x){
    ri f=1;c=gc();x=0;while(c<48||c>57){if(c=='-')f=-1;/*if(c==EOF){eof=0;break;}*/c
    =gc();}while(c>47&&c<58){x=(x<<3)+(x<<1)+(c^48);c=gc();}x*=f;}
}rd;


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

    return 0;
}
```

P4779 【模板】单源最短路径（标准版）

```
int dis[N],cnt=0,size=0,fr,tt,ww,n,m,s,head[N];

inline void add(int from,int to,int w){
    e[++cnt].w=w;
    e[cnt].v=to;
    e[cnt].nxt=head[from];
    head[from]=cnt;
}
```

```
inline void Push(node x){
    f[++size]=x;
    int now=size,nxt;
    while(now>1){
        nxt=now>>1;
        if(f[now].d>=f[nxt].d)return;
        swap(f[now],f[nxt]);
        now=nxt;
    }
}

inline void Del(){
    f[1]=f[size--];
    int now=1,nxt;
    while((now<<1)<=size){
        nxt=now<<1;
        if(nxt<size&&f[nxt].d>f[nxt+1].d)++nxt;
        if(f[nxt].d>=f[now].d)return;
        swap(f[now],f[nxt]);
        now=nxt;
    }
}

int main()
{
    freopen("read.txt","r",stdin);
    rd(n),rd(m),rd(s);
    rep(i,1,m){
        rd(fr),rd(tt),rd(ww);
        add(fr,tt,ww);
    }
    memm(dis);dis[s]=0;
    node tp;tp.d=0;tp.v=s;
    Push(tp);
    while(size>0){
        node qu=f[1];Del();
        int now=qu.v;
        if(dis[now]!=qu.d)continue;
        ee(i,now){
            int v=e[i].v,w=e[i].w;
            if(dis[now]+w<dis[v]){
                dis[v]=dis[now]+w;
                node tmp;tmp.v=v;tmp.d=dis[v];
                Push(tmp);
            }
        }
    }
    rep(i,1,n)wr(dis[i]);
    return 0;
```

```
}
```

P3379 【模板】最近公共祖先（LCA）

```
 struct EDGE{
    int v,nxt;
}e[N<<1];
int n,K,head[N],fa[N],cnt=0,dep[N],top[N],fr,tt,rt;

inline void add(ci from,ci to){
    e[++cnt].v=to;
    e[cnt].nxt=head[from];
    head[from]=cnt;
}

int dfs(int x,int far,int depth){
    dep[x]=depth;top[x]=x;int size=1,son=0,maxn=0;//这里 top[x]一定要赋初值为自己
    ee(i,x){
        int v=e[i].v;
        if(v==far)continue;
        fa[v]=x;
        int dx=dfs(v,x,depth+1);
        size+=dx;
        if(dx>maxn){
            son=v;
            maxn=dx;
        }
    }
    top[son]=x;
    return size;
}

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

int LCA(int x,int y){
    if(find(x)!=find(y)){
        return dep[top[x]]>dep[top[y]]?LCA(fa[top[x]],y):LCA(x,fa[top[y]]);
    }
    else return dep[x]>dep[y]?y:x;
}

int main()
{
//  freopen("read.txt","r",stdin);
    rd(n);rd(K);rd(rt);
    rep(i,1,n-1){
        rd(fr),rd(tt);
        add(fr,tt);
        add(tt,fr);
    }
```

```
    dfs(rt,0,0);
    rep(i,1,K){
        rd(fr),rd(tt);
        printf("%d\n",LCA(fr,tt));
    }
    return 0;
}
```

P3387 【模板】缩点（Tarjan,SCC)

```
int n,m,cnt=0,head[N],head2[N],cnt2=0,ans=-1,low[N],dfn[N],clk=0,co=0,top=1,stk[N],
dis[N],w[N],fa[N],ww[N];
bool instk[N];

struct EDGE{
    int v,nxt;
}e[M],e2[M];

inline void addedge(int from,int to)
{
    e[++cnt].v=to;
    e[cnt].nxt=head[from];
    head[from]=cnt;
}

void tarjan(int x)
{
    low[x]=++clk;dfn[x]=clk;
    instk[x]=true;stk[top++]=x;
    ee(x,i)
    {
        int v=e[i].v;
        if(!dfn[v])
        {
            tarjan(v);
            low[x]=mi(low[x],low[v]);
        }
        else if(instk[v])low[x]=mi(low[x],dfn[v]);
    }
    if(low[x]==dfn[x])
    {
        co++;
        do
        {
            fa[stk[--top]]=co;
            instk[stk[top]]=false;
            ww[co]+=w[stk[top]];
        }while(stk[top]!=x);
    }
}
```

```
void dp(int x)
{
    for(int i=head2[x];i;i=e2[i].nxt)
    {
        int v=e2[i].v;
        if(!dis[v])dp(v);
        dis[x]=ma(dis[x],dis[v]);
    }
    dis[x]=dis[x]+ww[x];
    ans=ma(ans,dis[x]);
    return;
}

void addedge2(int from,int to)
{
    e2[++cnt2].v=to;
    e2[cnt2].nxt=head2[from];
    head2[from]=cnt2;
}

int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m);mem(ww);
    rep(i,1,n)rd(w[i]);
    int fr,tt;
    mem(head);
    rep(i,1,m)
    {
       rd(fr),rd(tt);
       addedge(fr,tt);
    }
    mem(dfn);mem(instk);
    rep(i,1,n)if(!dfn[i])tarjan(i);
    mem(head2);
    rep(i,1,n)
    {
        ee(i,j)
        {
          int v=e[j].v;
          if(fa[i]!=fa[v])
          {
              addedge2(fa[i],fa[v]);
           }
        }
    }
    rep(i,1,co)if(!dis[i])dp(i);
    wr(ans);
```

```
    fwrite(sr,1,C+1,stdout);
    return 0;
}
```

P3388 【模板】割点（割顶）

```
void tarjan(int x,int fa)
{
    dfn[x]=++clk;low[x]=clk;
    child=0;
    ee(x)
    {
        int v=e[i].v;
        if(!dfn[v])
        {
            tarjan(v,x);
            low[x]=mi(low[x],low[v]);
            if(x!=fa&&low[v]>=dfn[x])cut[x]=true;
            if(x==fa)child++;
        }
        else low[x]=mi(low[x],dfn[v]);
    }
    if(x==fa&&child>=2)cut[x]=true;
}

int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m);
    mem(head);
    rep(i,1,m)
    {
       rd(fr),rd(tt);
       addedge(fr,tt);
       addedge(tt,fr);
    }
    mem(dfn);
    mem(cut);
    rep(i,1,n)
    {
        if(!dfn[i])tarjan(i,i);
    }
    rep(i,1,n)if(cut[i])ans++;
    prin(ans);
    rep(i,1,n)
    {
        if(cut[i])
        prik(i);
    }
```

```
    return 0;
}
```

P3367 【模板】并查集

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

void work()
{
    scanf("%d%d",&n,&m);
    for(int i=1;i<=n;i++)fa[i]=i;
    int tp,x,y;
    for(int i=1;i<=m;i++)
    {
       scanf("%d%d%d",&tp,&x,&y);
       if(tp==1){
          int rt=find(x);
          fa[rt]=find(y);
        }
       else if(find(x)==find(y))puts("Y");
       else puts("N");
    }
}
```

P3372 【模板】线段树 1

【标记永久化版】

```
void buildtree(int l,int r,int rt){
    if(l==r){
        sum[rt]=w[l];
        return;
    }
    int m=(l+r)>>1;
    buildtree(lson);
    buildtree(rson);
    pushup;
}

void updaterange(int x,int y,int dx,int l,int r,int rt){
    if(x<=l&&r<=y){
        //return sum[rt]+add[rt]*(r-l+1);
        add[rt]+=dx;
        return;
    }
    int m=(l+r)>>1;
    sum[rt]+=(mi(r,y)-ma(l,x)+1)*dx;
```

```
    if(x<=m)updaterange(x,y,dx,lson);
    if(y>m)updaterange(x,y,dx,rson);
}

ll queryrange(int x,int y,int l,int r,int rt){
    if(x<=l&&r<=y){
        return sum[rt]+add[rt]*(r-l+1);
    }
    int m=(l+r)>>1;
    ll res=(mi(r,y)-ma(x,l)+1)*add[rt];
    if(x<=m)res+=queryrange(x,y,lson);
    if(y>m)res+=queryrange(x,y,rson);
    return res;
}
```
【懒标记下传版】
```
void pushup(int rt)
{
    sum[rt]=sum[rt<<1]+sum[rt<<1|1];
}


void buildtree(int l,int r,int rt)
{
    add[rt]=0;
    if(l==r)
    {
        sum[rt]=a[l];
        return;
    }
    int m=(l+r)>>1;
    buildtree(l,m,rt<<1);
    buildtree(m+1,r,rt<<1|1);
    pushup(rt);
}

void updatenode(int l,int r,int rt,int k)
{
    add[rt]+=k;
    sum[rt]+=(r-l+1)*k;
}

void pushdown(int l,int r,int rt)
{
    if(!add[rt])return;
    int m=(l+r)>>1;
    updatenode(l,m,rt<<1,add[rt]);
    updatenode(m+1,r,rt<<1|1,add[rt]);
    add[rt]=0;
}
```

```
void update(int x,int y,int k,int l,int r,int rt)
{
    if(l>=x&&r<=y)
    {
        updatenode(l,r,rt,k);
        return;
    }
    pushdown(l,r,rt);
    int m=(l+r)>>1;
    if(x<=m)update(x,y,k,l,m,rt<<1);
    if(y>m)update(x,y,k,m+1,r,rt<<1|1);
    pushup(rt);
}

long long query(int x,int y,int l,int r,int rt)
{
    if(l>=x&&r<=y)return sum[rt];
    pushdown(l,r,rt);
    int m=(l+r)>>1;
    long long ans1=0,ans2=0;
    if(x<=m)ans1=query(x,y,l,m,rt<<1);
    if(y>m)ans2=query(x,y,m+1,r,rt<<1|1);
    return ans1+ans2;
}
```

【最值修改版（只能使用懒标记）】

```
#define pre int m=(l+r)>>1
#define pushup minn[rt]=min(minn[rt<<1],minn[rt<<1|1])
#define lson l,m,rt<<1
#define rson m+1,r,rt<<1|1
void buildtree(int l,int r,int rt){
    if(l==r){minn[rt]=a[l];return;}
    pre;buildtree(lson);buildtree(rson);pushup;
}

inline void updatenode(int rt,int k){lz[rt]+=k;minn[rt]+=k;return;}

void pushdown(int l,int r,int rt){
    if(!lz[rt])return;
    pre;updatenode(rt<<1,lz[rt]);updatenode(rt<<1|1,lz[rt]);lz[rt]=0;return;
}

void update(int x,int y,int k,int l,int r,int rt){
    if(x<=l&&r<=y){minn[rt]+=k;lz[rt]+=k;return;}
    pushdown(l,r,rt);pre;if(x<=m)update(x,y,k,lson);
    if(y>m)update(x,y,k,rson);pushup;
}

int query(int x,int y,int l,int r,int rt){
    if(x<=l&&r<=y)return minn[rt];
```

```
    pushdown(l,r,rt);pre;int ret=inf;
    if(x<=m)ret=min(ret,query(x,y,lson));if(y>m)ret=min(ret,query(x,y,rson));return
ret;
}
```

P3373 【模板】线段树 2

```
#define pre int m=(l+r)>>1
#define lson l,m,rt<<1
#define rson m+1,r,rt<<1|1
#define pushup sum[rt]=(sum[rt<<1]+sum[rt<<1|1])%p
void buildtree(int l,int r,int rt){
    if(l==r){
        sum[rt]=a[l]%p;add[rt]=0;mul[rt]=1;
        return;
    }
    pre;add[rt]=0;mul[rt]=1;
    buildtree(lson);buildtree(rson);pushup;//PUSH UP !!!
}

void pushdown(int l,int r,int rt){
    if(add[rt]==0&&mul[rt]==1)return;
    pre;
    sum[rt<<1]=(sum[rt<<1]*mul[rt]+add[rt]*(m-l+1))%p;
    sum[rt<<1|1]=(sum[rt<<1|1]*mul[rt]+add[rt]*(r-m))%p;
    add[rt<<1]=(add[rt<<1]*mul[rt]+add[rt])%p;
    add[rt<<1|1]=(add[rt<<1|1]*mul[rt]+add[rt])%p;
    mul[rt<<1]=mul[rt<<1]*mul[rt]%p;
    mul[rt<<1|1]=mul[rt<<1|1]*mul[rt]%p;
    add[rt]=0;mul[rt]=1;return;
}

void update1(int x,int y,ll k,int l,int r,int rt){
    if(x<=l&&r<=y){
        sum[rt]=(sum[rt]*k)%p;
        add[rt]=(add[rt]*k)%p;
        mul[rt]=(mul[rt]*k)%p;
        return;
    }
    pre;pushdown(l,r,rt);
    if(x<=m)update1(x,y,k,lson);
    if(y>m)update1(x,y,k,rson);
    pushup;
}

void update2(int x,int y,ll k,int l,int r,int rt){
    if(x<=l&&r<=y){
        //add[rt]+=k;add[rt]%=p;
        add[rt]=(add[rt]+k)%p;
```

```
        sum[rt]=(sum[rt]+k*(r-l+1))%p;
        return;
    }
    pre;pushdown(l,r,rt);
    if(x<=m)update2(x,y,k,lson);
    if(y>m)update2(x,y,k,rson);
    pushup;
}

ll query(int x,int y,int l,int r,int rt){
    if(x<=l&&r<=y)return sum[rt];
    pre;ll ret=0;
    pushdown(l,r,rt);
    if(x<=m)ret=(ret+query(x,y,lson))%p;
    if(y>m)ret=(ret+query(x,y,rson))%p;
    pushup;
    return ret;
}
```

P3366 【模板】最小生成树

【Kruskal】

```
int n,m,cnt=0,ans=0,fa[N];
struct EDGE{
   int v1,v2,w;
}e[M];
struct cmp{
   inline bool operator()(const EDGE&T1,const EDGE&T2){
      return T1.w<T2.w;
   }
};
int find(int x){return fa[x]==x?x:fa[x]=find(fa[x]);}

int main()
{
    freopen("read.txt","r",stdin);
    rd>>n>>m;
    rep(i,1,m)rd>>e[i].v1>>e[i].v2>>e[i].w;
    sort(e+1,e+m+1,cmp());
    rep(i,1,n)fa[i]=i;
   rep(i,1,m){
      int v1=e[i].v1,v2=e[i].v2;
      int r1=find(v1),r2=find(v2);
      if(r1!=r2){
         fa[r1]=r2;
         ans+=e[i].w;
         ++cnt;
         if(cnt==n-1)break;
      }
   }
```

```
    if(cnt==n-1){
        printf("%d",ans);
    }
    else puts("orz");
    return 0;
}
```
【Prim <->久了不打确实不熟悉啊，它容易和最短路混淆】

```
int head[N],cnt=0,hv=0,ans=0,vis[N],n,m,fr,tt,ww;
struct EDGE{
    int v,w,nxt;
}e[M<<1];
struct NODE{
    int d,v;//d表示已联通集合到其它点的距离
    bool operator<(const NODE&rhs)const{
        return d>rhs.d;//优先队列注意反转,我也是神了，把d写成v都能过样例
    }
};
inline void add(ci from,ci to,ci w){
    e[++cnt].v=to;
    e[cnt].nxt=head[from];
    e[cnt].w=w;
    head[from]=cnt;
}
priority_queue<NODE> q;

int main()
{
    freopen("read.txt","r",stdin);
    rd>>n>>m;
    rep(i,1,m){
        rd>>fr>>tt>>ww;
        add(fr,tt,ww);
        add(tt,fr,ww);
    }
    NODE sts;sts.d=0;sts.v=1;q.push(sts);
    while(!q.empty()){
        NODE qu=q.top();q.pop();
        if(vis[qu.v])continue;
        ans+=qu.d;
        vis[qu.v]=1;
        if(++hv==n)break;
        ee(i,qu.v){
            int v=e[i].v;
            if(vis[v])continue;
            NODE ns;ns.d=e[i].w;ns.v=v;q.push(ns);
        }
    }
    if(hv==n)printf("%d",ans);
    else puts("orz");
```

```
    return 0;
}
```

P3865 【模板】ST 表

```
int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m);
    rep(i,1,n)rd(a[i][0]),lg[i]=lg[i-1]+(1<<lg[i-1]==i);
    rep(j,1,lg[n]-1)
    {
       for(ri i=1;i+(1<<j)-1<=N;i++)
       {
          a[i][j]=max(a[i][j-1],a[i+(1<<(j-1))][j-1]);
        }
    }
    int l,r,k,ans;
    rep(i,1,m)
    {
        rd(l),rd(r);
        k=lg[r-l+1]-1;
        ans=max(a[l][k],a[r-(1<<k)+1][k]);
        wr(ans);
    }
    fwrite(sr,1,C+1,stdout);
    return 0;
}
```

P1198 [JSOI2008]最大数(逆向ST表，可用于动态加点查询最值)

```
void update(ll k){
   f[n][0]=k;
   for(int i=1;n-(1<<i)+1>=1;i++)f[n][i]=max(f[n][i-1],f[n-(1<<(i-1))][i-1]);
}

ll query(int l,int r){
   int K=log(r-l+1)/log(2);
   return max(f[r][K],f[l+(1<<K)-1][K]);
}


int main()
{
    //freopen("read.txt","r",stdin);
    rd(m);rd(mod);
    rep(i,1,m){
       while(c!='A'&&c!='Q')c=gc();
       if(c=='A'){
          rd(xx);++n;
          update((xx+t)%mod);
       }
```

```
        if(c=='Q'){
            rd(xx);
            t=query(n-xx+1,n);
            printf("%lld\n",t);
        }
    }
    return 0;
}
```

P3374 【模板】树状数组 1

```
inline int lowbit(int x){return x&(-x);}

ll Sum(int x){
    int ret=0;
    for(int i=x;i;i-=lowbit(i))ret+=a[i];
    return ret;
}


int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m);
    rep(i,1,n){
        rd(tp);
        for(int j=i;j<=n;j+=lowbit(j))a[j]+=tp;
    }
    rep(i,1,m){
        rd(op);
        if(op==1){
            rd(x);rd(dx);
            for(int j=x;j<=n;j+=lowbit(j))a[j]+=dx;
        }
        else{
            rd(x);rd(y);
            printf("%lld\n",Sum(y)-Sum(x-1));
        }
    }
    return 0;
}
```

P3368 【模板】树状数组 2

```
inline int lowbit(int x){return x&(-x);}

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

```
    rd(n);rd(m);
    rep(i,1,n){
        rd(a[i]);
        //for(int j=x;j<=n;j+=lowbit(j))a[j]+=x;
    }
    rep(i,1,m){
        rd(op);
        if(op==1){
            rd(x);rd(y);rd(dx);
            for(int i=x;i<=n;i+=lowbit(i))cf[i]+=dx;
            for(int i=y+1;i<=n;i+=lowbit(i))cf[i]+=-dx;
        }
        else{
            rd(x);
            dx=0;
            for(int i=x;i;i-=lowbit(i))dx+=cf[i];
            printf("%lld\n",dx+a[x]);
        }
    }
    return 0;
}
```

【拓展】树状数组 3 （区间修改，区间查询）

```
ll d1[N],d2[N],n,m,dx,x,y,op,lst;
inline int lowbit(int x){return x&(-x);}

void update(int x,int y,ll dx){
    ++y;
    for(int i=x;i<=n;i+=lowbit(i)){
        d1[i]+=dx;
        d2[i]+=dx*x;
    }
    for(int i=y;i<=n;i+=lowbit(i)){
        d1[i]-=dx;
        d2[i]-=dx*y;
    }
}

ll query(int x,int y){
    ll ret=0;
    for(int i=x-1;i;i-=lowbit(i)){
        ret-=x*d1[i]-d2[i];
    }
    for(int i=y;i;i-=lowbit(i)){
        ret+=(y+1)*d1[i]-d2[i];
    }
    return ret;
}
```

```
int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m);lst=0;
    rep(i,1,n){
        rd(x);
        dx=x-lst;
        lst=x;
        for(int j=i;j<=n;j+=lowbit(j)){
            d1[j]+=dx;
            d2[j]+=i*dx;
        }
    }
    rep(i,1,m){
        rd(op);
        if(op==1){
            rd(x),rd(y),rd(dx);
            update(x,y,dx);
        }
        else{
            rd(x),rd(y);
            printf("%lld\n",query(x,y));
        }
    }
    return 0;
}
```

P3384 【模板】树链剖分

【1.第二次交换 x 和 y 是使 x 深度较小 2.第一次交换是 dep[top[x]]与 dep[top[y]]】3.记得随时取模 4.区间是 r-l+1】

```
struct EDGE{
    int v,nxt;
}e[N<<1];
int cnt=0,dep[N],head[N],size[N],top[N],fa[N],son[N],id[N],neww[N],w[N],sum[N<<2],mod,add[N<<2],n,m,root,fr,tt,op;


inline void addedge(int from,int to)
{
    e[++cnt].v=to;
    e[cnt].nxt=head[from];
    head[from]=cnt;
}

void dfs1(int x,int deep)
{
    dep[x]=deep;size[x]=1;int maxn=0,tpson=0;
    ee(i,x){
```

```
        int v=e[i].v;
        if(v==fa[x])continue;
        fa[v]=x;
        dfs1(v,deep+1);
        size[x]+=size[v];
        if(size[v]>maxn){
            maxn=size[v];
            tpson=v;
        }
    }
    if(tpson)son[x]=tpson;
}

void dfs2(int x,int topp)
{
    id[x]=++cnt;neww[id[x]]=w[x];top[x]=topp;
    if(son[x])dfs2(son[x],topp);
    else return;
    ee(i,x){
        int v=e[i].v;
        if(v==fa[x]||v==son[x])continue;
        dfs2(v,v);
    }
}

void buildtree(int l,int r,int rt)
{
    if(l==r){
        sum[rt]=neww[l]%mod;
        return;
    }
    int m=(l+r)>>1;
    buildtree(lson);
    buildtree(rson);
    pushup;
}

inline void updatenode(int dx,int l,int r,int rt){
    sum[rt]+=(r-l+1)*dx;
    sum[rt]%=mod;
    add[rt]+=dx;
    add[rt]%=mod;
}

inline void pushdown(int l,int r,int rt)
{
    if(!add[rt])return;
    int m=(l+r)>>1;
    updatenode(add[rt],lson);
    updatenode(add[rt],rson);
```

```
    add[rt]=0;
}

void updaterange(int x,int y,int dx,int l,int r,int rt)
{
    if(x<=l&&r<=y){
        add[rt]+=dx;
        add[rt]%=mod;
        sum[rt]+=(r-l+1)*dx;//<<!!!
        sum[rt]%=mod;
        return;
    }
    pushdown(l,r,rt);
    int m=(l+r)>>1;
    if(x<=m)updaterange(x,y,dx,lson);
    if(y>m)updaterange(x,y,dx,rson);
    pushup;
}

void Update(int x,int y,int dx)
{
    while(top[x]!=top[y]){
        if(dep[top[x]]<dep[top[y]])swap(x,y);
        updaterange(id[top[x]],id[x],dx,1,n,1);
        x=fa[top[x]];
    }
    if(dep[x]>dep[y])swap(x,y);
    updaterange(id[x],id[y],dx,1,n,1);
}

int queryrange(int x,int y,int l,int r,int rt)
{
    if(x<=l&&r<=y){
        return sum[rt]%mod;
    }
    int ans=0,m=(l+r)>>1;
    pushdown(l,r,rt);
    if(x<=m)ans+=queryrange(x,y,lson)%mod;
    if(y>m)ans=(ans+queryrange(x,y,rson))%mod;
    return ans;
}

int Query(int x,int y)
{
    int ans=0;
    while(top[x]!=top[y]){
        if(dep[top[x]]<dep[top[y]]){
            swap(x,y);
        }
        ans+=queryrange(id[top[x]],id[x],1,n,1);
```

```
        ans%=mod;
        x=fa[top[x]];
    }
    if(dep[x]>dep[y]){
        swap(x,y);
    }
    ans+=queryrange(id[x],id[y],1,n,1);
    return ans%mod;
}


int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m),rd(root),rd(mod);
    rep(i,1,n)rd(w[i]);
    rep(i,1,n-1){
        rd(fr),rd(tt);
        addedge(fr,tt);
        addedge(tt,fr);
    }
    cnt=0;
    dfs1(root,0);
    dfs2(root,root);
    buildtree(1,n,1);
    rep(i,1,m){
        rd(op);
        if(op==1){
            rd(fr),rd(tt),rd(tp);
            Update(fr,tt,tp);
        }
        else if(op==2){
            rd(fr),rd(tt);
            wr(Query(fr,tt));
        }
        else if(op==3){
            rd(fr),rd(tp);
            updaterange(id[fr],id[fr]+size[fr]-1,tp,1,n,1);
        }
        else if(op==4){
            rd(tp);
            wr(querytrange(id[tp],id[tp]+size[tp]-1,1,n,1));
        }
    }
    return 0;
}
```

P1226 【模板】快速幂||取余运算

```
int main()
```

```
{
    scanf("%lld%lld%lld",&b,&p,&k);
    t1=b,t2=p;
    while(p)
    {
        if(p&1)ans=ans*b%k;
        b=b*b%k;
        p>>=1;
    }
    printf("%lld^%lld mod %lld=%lld",t1,t2,k,ans%k);
    return 0;
}
```

P3383 【模板】线性筛素数

```
int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m);
    not_prime[1]=true;
    rep(i,2,n)
    {
        if(!not_prime[i])prime[++cnt]=i;
        for(ri j=1;i*prime[j]<=n;j++)
        {
            not_prime[prime[j]*i]=true;
            if(i%prime[j]==0)break;
        }
    }
    int x;
    rep(i,1,m)
    {
        rd(x);
        if(not_prime[x])puts("No");
        else puts("Yes");
    }
    return 0;
}
```

P1020 导弹拦截【LIS+不定项数据的快速读入】

```
/*
Problem:P1020 复习，首次使用快读读取不定项数据，复习最长上升子序列(lower_bound)，最长不下降子序列(upper_bound)，最长下降子序列，最长不上升子序列（后两个把前两个倒着求救可以了）
User:mzg1816_LRL
Date:2019.2.8
Algorithm:LIS,lower_bound,upper_bound 的妙用
*/
#define ri register int
```

```
#define ll long long
#define ci const int&
#define rep(i,a,b) for(int i=(a);i<=(b);++i)
#define dwn(i,a,b) for(int i=(a);i>=(b);--i)
#define ee(i,a) for(int i=head[a];i;i=e[i].nxt)
#include<bits/stdc++.h>
using namespace std;
const int N=100055;
const int M=10055;
const int inf=0x3f3f3f3f;
char ss[1<<21],*A=ss,*B=ss,c;int eof=1,n=1;
inline char gc(){return A==B&&(B=(A=ss)+fread(ss,1,1<<21,stdin),A==B)?EOF:*A++;}
template<class T>inline void rd(T&x){
    ri f=1;c=gc();x=0;while(c<48||c>57){/*if(c=='-')f=-1;*/if(c==EOF){eof=0;return;}c=gc();}
    while(c>47&&c<58){x=(x<<3)+(x<<1)+(c^48);c=gc();}x*=f;++n;//一定要在这里（成功读入后）对 n 计数，才不会有 bug
}
int up[N],dw[N],f1[N],f2[N];

int main()
{
// freopen("read.txt","r",stdin);
    while(c!=EOF){if(eof){rd(up[n]);}}--n;//快读照样飞起，可以完美处理文件末尾有无换行符的情况，n 初值赋的 1，要减一
    rep(i,1,n)dw[n-i+1]=up[i];//序列翻转,其实不反转也可以，在处理的时候翻转是一样的，不过这样可以提高可读性
    memset(f1,0x3f,sizeof(f1));//预处理
    memset(f2,0x3f,sizeof(f2));
    rep(i,1,n){
        *upper_bound(f1+1,f1+n+1,dw[i])=dw[i];//求解最长不上升子序列，即第一问答案
        *lower_bound(f2+1,f2+n+1,up[i])=up[i];//求解最长上升子序列，即第二问答案
    }
    printf("%lld\n%lld",lower_bound(f1+1,f1+n+1,inf)-f1-1,lower_bound(f2+1,f2+n+1,inf)-f2-1);//输出答案，由于 inf 是非法位置，恰在最长序列的右端+1 的位置，所以要-1
    return 0;
}
```

P3807 【模板】卢卡斯定理

```
ll C(ll x,ll y){
    if(y>x)return 0;
    return a[x]*b[y]%p*b[x-y]%p;//组合数计算，除以一个数等价于乘它的逆元
}

ll lucas(ll x,ll y){
    if(x<=p&&y<=p)return C(x,y);
    return lucas(x/p,y/p)*lucas(x%p,y%p)%p;//Lucas 定理
}
```

```
int main()
{
   // freopen("read.txt","r",stdin);
    int T;rd(T);
    while(T--){
       rd(n);rd(m);rd(p);
       a[0]=a[1]=1;//a[0]=b[0]=1 很重要!!!
       rep(i,2,p)a[i]=a[i-1]*i%p;//阶乘数组
       inv[1]=1;
       rep(i,2,p)inv[i]=(p-p/i)*inv[p%i]%p;//线性递推逆元数组（要求 p 为质数）
       b[0]=b[1]=1;
       rep(i,2,p)b[i]=b[i-1]*inv[i]%p;//逆元前缀积
       printf("%lld\n",lucas(n+m,m));
    }
    return 0;
}
```

P3811 【模板】乘法逆元【包括 GCD,EDGCD】

【1.inv 数组一定要开 long long】

```
int qp(int a,int k){
   ll ret=1,dx=(ll)a;
   while(k){
      if(k&1)ret=ret*dx%p;
      dx=dx*dx%p;
      k>>=1;
   }
   return (int)ret;
}

int exgcd(int a,int b,int &x,int &y){
   if(a==0){
      x=0;y=1;return b;
   }
   int d=exgcd(b%a,a,y,x);
   x-=b/a*y;
   return d;
}

int main()
{
   freopen("read.txt.","r",stdin);
   rd(n);rd(p);
   //快速幂求逆元
   rep(i,1,n){
      printf("%d\n",qp(i,p-2)%p);
   }
   putchar('\n');
   //扩展欧几里得 Exgcd 求逆元（求得 ax+py=1 的一组 x>0 的解，则 x%p 即为所求）
   rep(i,1,n){
```

```
        exgcd(i,p,X,Y);
        printf("%d\n",(X%p+p)%p);
    }
    putchar('\n');

    //线性递推逆元
    inv[1]=1;//inv 数组一定要开 long long
    rep(i,2,n)inv[i]=(p-p/i)*inv[p%i]%p;
    rep(i,1,n)printf("%lld\n",inv[i]);
    return 0;
}
```

P3868 [TJOI]猜数字 中国剩余定理（CRT）

```
ll mul(ll a,ll b){
    ll ret=0,dx=a;
    while(b){
        if(b&1)ret=(ret+dx)%M;
        dx<<=1;
        dx%=M;
        b>>=1;
    }
    return ret;
}

int main()
{
    //freopen("read.txt","r",stdin);
    rd(n);
    rep(i,1,n)rd(ai[i]);
    rep(i,1,n){
        rd(bi[i]);
        M*=bi[i];
        ai[i]=(ai[i]%bi[i]+bi[i])%bi[i];
    }
    rep(i,1,n){
        ll Mi=M/bi[i];
        exgcd(Mi,bi[i],x,y);
        ll inv=(x%bi[i]+bi[i])%bi[i];
        Mi=mul(mul(Mi,inv),ai[i]);
        ans=(ans+Mi)%M;
    }
    printf("%lld",ans);
    return 0;
}
```

P4777 【模板】扩展中国剩余定理（EXCRT）

```
int main()
```

```
{
    //freopen("read.txt","r",stdin);
    rd(n);
    rep(i,1,n){
        rd(bi[i]);rd(ai[i]);
    }
    ll ans=ai[1],M=bi[1];
    rep(i,2,n){
        ll a=M,b=bi[i],c=((ai[i]-ans)%b+b)%b;
        ll d=exgcd(a,b,x,y);
        if(c%d)return 0;//无解
        x=mul(x,c/d,b/d);//模数为什么是b/d,反正这样对答案无影响(推了我一下午),与这个有关:ax≡c(mod b)
        ans+=x*M;
        M*=b/d;//本题b/d妙用很多,这里可以求最小公倍数,因为lcm(M,b[i])=M*b[i]/gcd
        ans=(ans%M+M)%M;
    }
    printf("%lld",ans);
    return 0;
}
```

P2613 【模板】有理数取余

```
template<class T>inline void re(T&x){x=(x%mod+mod)%mod;}
int main()
{
  //  freopen("read.txt","r",stdin);
    ll a,b;
    rd(a);rd(b);
    if(!b){
        puts("Angry!");
        return 0;
    }
    ll x,y;
    exgcd(b,mod,x,y);
    re(x);
    printf("%lld",x*a%mod);
    return 0;
}
```

P5091 【模板】欧拉定理

```
int main()
{
    //freopen("read.txt","r",stdin);
    rd(a);rd(mod);
    ll tmp=mod;phi=mod;
    for(int i=2;i*i<=mod;++i){
        if(tmp%i==0){
```

```
            phi=phi-phi/i;
            while(tmp%i==0)tmp/=i;
        }
    }
    if(tmp>1)phi=phi-phi/tmp;
    ll b=0;
    while(!isdigit(c))c=gc();
    while(isdigit(c)){
        b=(b<<1)+(b<<3)+(c^48);
        if(b>=phi){
            ok=1;
            b%=phi;//对欧拉函数取余，不是 mod!!!
        }
        c=gc();
    }
    if(ok)b+=phi;
    while(b){
        if(b&1)ans=ans*a%mod;
        a=a*a%mod;
        b>>=1;
    }
    printf("%lld",ans);
    return 0;
}
```

P3390 【模板】矩阵快速幂

```
int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(kk);
    rep(i,1,n){
       rep(j,1,n){
          rd(a[i][j]);
          if(i==j)ans[i][j]=1;
        }
    }
    while(kk)
    {
       if(kk&1){
          rep(i,1,n){
             rep(j,1,n){
                tp[i][j]=0;
              }
          }
          rep(i,1,n){
              rep(j,1,n){
                  rep(k,1,n){
                      tp[i][j]+=a[i][k]*ans[k][j];
                      tp[i][j]%=mod;
```

```
                }
            }
        }
        rep(i,1,n){
          rep(j,1,n){
             ans[i][j]=tp[i][j];
            }
        }
    }
    rep(i,1,n){
      rep(j,1,n){
         tp[i][j]=0;
        }
    }
    rep(i,1,n){
        rep(j,1,n){
            rep(k,1,n){
                tp[i][j]+=a[i][k]*a[k][j];
                tp[i][j]%=mod;
            }
        }
    }
    rep(i,1,n){
      rep(j,1,n){
         a[i][j]=tp[i][j];
        }
    }
    kk>>=1;
}
rep(i,1,n){
    rep(j,1,n){
        wrl(ans[i][j]);
        putchar(' ');
    }
    putchar('\n');
}
return 0;
}
```

P3375 【模板】KMP 字符串匹配

```
int main()
{
    //freopen("read.txt","r",stdin);
    scanf("%s",A+1);
    scanf("%s",B+1);
    n=strlen(A+1);m=strlen(B+1);
    int j=0;p[1]=0;
    rep(i,2,m)
    {
```

```
        while(j&&B[j+1]!=B[i])j=p[j];
        if(B[j+1]==B[i])++j;
        p[i]=j;
    }
    j=0;
    rep(i,1,n)
    {
        while(j&&A[i]!=B[j+1])j=p[j];
        if(B[j+1]==A[i])++j;
        if(j==m)
        {
            wr(i-m+1);
            //++ans;
            putchar('\n');
            j=p[j];
        }
    }
    //wr(ans);
    rep(i,1,m)wr(p[i]),putchar(' ');
    return 0;
}
```

P1439 【模板】最长公共子序列

```
rd(n);mem(p);
    rep(i,1,n)
    {
        rd(x);
        if(!p[x])pos[x].clear();
        pos[x].push_back(i);p[x]++;
    }
    memm(f);
    rep(i,1,n)
    {
        rd(x);
        for(ri j=p[x]-1;j>=0;j--)
        {
            *lower_bound(f,f+n,pos[x][j])=pos[x][j];
        }
    }
    wr(lower_bound(f,f+n,0x3f3f3f3f)-f);
```

P3808 【模板】AC 自动机

```
    int main()
    {
        //freopen("read.txt","r",stdin);
        scanf("%d",&n);
        rep(i,1,n)
```

```
        {
          scanf("%s",s);//insert_s
          int len=strlen(s),p=0;
          rep(j,0,len-1)
          {
             int x=s[j]-'a';
             if(trie[p][x]==0)trie[p][x]=++cnt;
             p=trie[p][x];
           }
           ++num[p];
        }
        //build_nxt_function
        rep(i,0,25)if(trie[0][i])nxt[trie[0][i]]=0,q.push(trie[0][i]);
        while(!q.empty())
        {
            int now=q.front();q.pop();
            rep(i,0,25)
            {
                if(trie[now][i])nxt[trie[now][i]]=trie[nxt[now]][i],q.push(trie[now]
[i]);
                else trie[now][i]=trie[nxt[now]][i];
            }
        }
        scanf("%s",s);
        int len=strlen(s),p=0;
        rep(i,0,len-1)
        {
            p=trie[p][s[i]-'a'];
            for(int j=p;j&&~num[j];j=nxt[j])ans+=num[j],num[j]=-1;
        }
        wr(ans);
        return 0;
    }
```

P3376 【模板】网络最大流

```
    bool bfs()
    {
        memf(dis);
        dis[sts]=0;
        queue<int> q;
        q.push(sts);
        while(!q.empty())
        {
            int nw=q.front();q.pop();
            ee(i,nw)
            {
                int v=e[i].v;
                if(dis[v]==-1&&e[i].w>0)
                {
```

```
                dis[v]=dis[nw]+1;
                q.push(v);
            }
        }
    }
    return dis[ed]!=-1;
}

int max_flow(int now,int flow)
{
    if(now==ed)return flow;
    int tp=flow;
    for(int &i=cur[now];i+1;i=e[i].nxt)
    {
        int v=e[i].v,w=e[i].w;
        if(dis[v]==dis[now]+1&&w)
        {
            int dx=max_flow(v,mi(tp,w));
            e[i].w-=dx;e[i^1].w+=dx;
            tp-=dx;
            if(tp==0)break;
        }
    }
    return flow-tp;
}


int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m),rd(sts),rd(ed);
    memf(head);
    rep(i,1,m)
    {
      rd(fr),rd(tt),rd(ww);
      addedge(fr,tt,ww);
      addedge(tt,fr,0);
    }
    while(bfs())
    {
        rep(i,1,n)cur[i]=head[i];
        ans+=max_flow(sts,inf);
    }

        wr(ans);
        return 0;
}
```

P3381 【模板】最小费用最大流

```
bool SPFA()
{
    memm(dis);mem(vis);dis[sts]=0;
    queue<int> q;q.push(sts);
    while(!q.empty())
    {
        int nw=q.front();q.pop();vis[nw]=false;
        for(int i=head[nw];i+1;i=e[i].nxt)
        {
            int v=e[i].v;
            if(e[i].w&&dis[nw]+e[i].fee<dis[v])
            {
                dis[v]=dis[nw]+e[i].fee;
                if(!vis[v])
                {
                    q.push(v);
                    vis[v]=true;
                }
            }
        }
    }
    return dis[ed]!=0x3f3f3f3f;
}


int max_flow(int u,int flow)
{
    if(u==ed)return flow;
    int tp=flow;
    vis[u]=true;
    for(int &i=cur[u];i+1;i=e[i].nxt)
    {
        int v=e[i].v;
        if(!vis[v]&&e[i].w&&dis[v]==dis[u]+e[i].fee)
        {
            //printf("#%d",e[i].fee);
            int dx=max_flow(v,mi(flow,e[i].w));
            minfee+=dx*e[i].fee;
            //minfee+=dx*e[i^1].fee;
            e[i].w-=dx;
            e[i^1].w+=dx;
            flow-=dx;
            if(flow==0)break;
        }
    }
    vis[u]=false;
    return tp-flow;
}
```

```
int main()
{
    //freopen("read.txt","r",stdin);
    rd(n),rd(m),rd(sts),rd(ed);
    memf(head);
    rep(i,1,m)
    {
      rd(fr),rd(tt),rd(ww),rd(fei);
      addedge(fr,tt,ww,fei);
      addedge(tt,fr,0,-fei);
    }
    while(SPFA())
    {
        rep(i,1,n)cur[i]=head[i];
        mem(vis);
        maxn+=max_flow(sts,inf);
    }
    wr(maxn);
    wr(minfee);
    return 0;
}
```

2019.2.10 v1.0
By LRL