

[back](#)

```
#include<bits/stdc++.h>
using namespace std;
const int N=100005;
typedef long long ll;
int n,m,p;
ll tr[N*4],mul[N*4],add[N*4],asi[N*4],a[N];
inline int ls(int o){
    return o<<1;
}
inline int rs(int o){
    return o<<1|1;
}
void push_up(int o){
    tr[o]=tr[ls(o)]+tr[rs(o)];
}
void build(int o,int l,int r){
    mul[o]=1;
    asi[o]=-1;
    if(l==r){
        tr[o]=a[l];
        return;
    }
    int mid=l+r>>1;
    build(ls(o),l,mid);
    build(rs(o),mid+1,r);
    push_up(o);
}
void f(int o,int l,int r,ll k){
    tr[o]=(tr[o]+(r-l+1)*k)%p;
    add[o]=(add[o]+k)%p;
}
void g(int o,int l,int r,ll k){
    tr[o]=tr[o]*k%p;
    mul[o]=mul[o]*k%p;
    add[o]=add[o]*k%p;
}
void h(int o,int l,int r,ll k){
    add[o]=0;
    mul[o]=1;
    asi[o]=k;
    tr[o]=k*(r-l+1)%p;
}
void push_down(int o,int l,int r){
    int mid=l+r>>1;
    if(asi[o]!=-1){
        h(ls(o),l,mid,asi[o]);
        h(rs(o),mid+1,r,asi[o]);
        asi[o]=-1;
    }
}
```

```
}
if(mul[o]!=1){
    g(ls(o),l,mid,mul[o]);
    g(rs(o),mid+1,r,mul[o]);
    mul[o]=1;
}
if(add[o]){
    f(ls(o),l,mid,add[o]);
    f(rs(o),mid+1,r,add[o]);
    add[o]=0;
}
}

void xg(int o,int nl,int nr,int l,int r,ll k,int op){
    if(nl<=l&&r<=nr){
        if(op==1){
            mul[o]=mul[o]*k%p;
            add[o]=add[o]*k%p;
            tr[o]=tr[o]*k%p;
        }
        else if(op==2){
            add[o]=(add[o]+k)%p;
            tr[o]=(tr[o]+(r-l+1)*k)%p;
        }
        else if(op==3){
            add[o]=0;
            mul[o]=1;
            tr[o]=k*(r-l+1)%p;
            asi[o]=k;
        }
        return;
    }
    push_down(o,l,r);
    int mid=l+r>>1;
    if(nl<=mid) xg(ls(o),nl,nr,l,mid,k,op);
    if(nr>mid) xg(rs(o),nl,nr,mid+1,r,k,op);
    push_up(o);
}

ll cx(int o,int nl,int nr,int l,int r){
    if(nl<=l&&r<=nr) return tr[o];
    push_down(o,l,r);
    int mid=l+r>>1;
    ll ret=0;
    if(nl<=mid) ret=(ret+cx(ls(o),nl,nr,l,mid))%p;
    if(nr>mid) ret=(ret+cx(rs(o),nl,nr,mid+1,r))%p;
    return ret;
}

int main(){
    scanf("%d %d %d",&n,&m,&p);
    for(int i=1;i<=n;i++) scanf("%lld",&a[i]);
    build(1,1,n);
}
```

```
for(int i=1;i<=m;i++){
    int op,x,y;
    scanf("%d %d %d",&op,&x,&y);
    if(op==1||op==2||op==3){
        ll k;
        scanf("%lld",&k);
        xg(1,x,y,1,n,k,op);
    }
    else printf("%lld\n",cx(1,x,y,1,n));
}
return 0;
}
```

From:

<https://wiki.cvbbacm.com/> - **CVBB ACM Team**

Permanent link:

https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:code3



Last update: **2020/05/07 11:13**