

```
<code>

#include<bits/stdc++.h> using namespace std; const int N=500005; typedef long long ll; ll
tr[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){

    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 xg(int o,int pos,int l,int r,ll k,int op){

    if(l==r){
        if(op==1) tr[o]+=k;
        else tr[o]=k;
        return;
    }
    int mid=l+r>>1;
    if(pos<=mid) xg(ls(o),pos,l,mid,k,op);
    else xg(rs(o),pos,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];
    int mid=l+r>>1;
    ll ret=0;
    if(nl<=mid) ret+=cx(ls(o),nl,nr,l,mid);
    if(nr>mid) ret+=cx(rs(o),nl,nr,mid+1,r);
    return ret;

} int main(){

    int n,m;
```

```
scanf("%d %d",&n,&m);
for(int i=1;i<=n;i++){
    scanf("%d",&a[i]);
}
build(1,1,n);
for(int i=1;i<=m;i++){
    int op;
    scanf("%d",&op);
    if(op==1||op==3){
        int pos;
        ll k;
        scanf("%d %lld",&pos,&k);
        xg(1,pos,1,n,k,op);
    }
    else{
        int l,r;
        scanf("%d %d",&l,&r);
        printf("%lld\n",cx(1,l,r,1,n));
    }
}
return 0;
}
```

<\code>

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



Permanent link:
https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:code&rev=1588820971

Last update: **2020/05/07 11:09**