```cpp
#include <bits/stdc++.h>
using namespace std;

#define mp make pair
#define F first
#define S second
typedef long long int ll;
typedef pair<ll,ll> ii;
typedef vector<ll> vi;
typedef vector<ii > vii;

vi A;
vii st;
int n,m;

int getMax(vector<ll> e){
    ll s = -1;
    int p=-1;
    for(int i=0;i<e.size();i++){
        if(e[i]!=-1 and s < A[e[i]]){
            s = A[e[i]];
            p = i;
        }
    }
    return p;
}

void build(int P,int L,int R){
    if(L==R){
        st[P] = mp(L,-1);
        return;
    }
    if(L>R || R<L)
        return;
    int nxt = P << 1;
    int mid = (L+R) >> 1;
    build(nxt,L,mid);
    build(nxt+1,mid+1,R);

    vi e;
    ll s1,s2;
    int p;
    e.push back(st[nxt].F);
    e.push back(st[nxt].S);
    e.push back(st[nxt+1].F);
    e.push back(st[nxt+1].S);
    p = getMax(e);
    s1 = p==-1 ? -1:e[p];
    e[p]=-1;
    p = getMax(e);
    s2 = p==-1 ? -1:e[p];

    st[P] = mp(s1,s2);

}

void update (int p, int L, int R, int i, int value) {

    // no overlap
    if(L > i or R < i) return;

    // total overlap
    if(L == R and L == i) {
        A[i] = value;
        st[p] = mp(i,-1);
        return;
    }

    int nxt = p << 1;
    int mid = (L + R) >> 1;
```

```
71      update (nxt, L, mid, i, value);
72      update (nxt + 1, mid + 1, R, i, value);
73
74      vi e;
75      ll s1,s2;
76      int V;
77      e.push back(st[nxt].F);
78      e.push back(st[nxt].S);
79      e.push back(st[nxt+1].F);
80      e.push back(st[nxt+1].S);
81      V = getMax(e);
82      s1 = V==-1? -1:e[V];
83      e[V]=-1;
84      V = getMax(e);
85      s2 = V==-1? -1:e[V];
86      st[p].F = s1;
87      st[p].S = s2;
88  }
89  ii query(int p, int L, int R, int i, int j){
90      // no overlap
91      if(i>R || j<L) return mp(-1,-1);
92
93      // total overlap
94      if(L>=i && R<=j) return st[p];
95
96      // partial overlap
97      int nxt = p << 1;
98      int mid = (L + R) >> 1;
99      ii p1 = query(nxt,L,mid,i,j);
100     ii p2 = query(nxt + 1,mid +1,R,i,j);
101
102     if(p1.F==-1 and p1.S==-1) return p2;
103     if(p2.F==-1 and p2.S==-1) return p1;
104
105     vi e;
106     ll s1,s2;
107     int V;
108     e.push back(p1.F);
109     e.push back(p1.S);
110     e.push back(p2.F);
111     e.push back(p2.S);
112     V = getMax(e);
113     s1 = V==-1? -1:e[V];
114     e[V] = -1;
115     V = getMax(e);
116     s2 = e[V];
117
118     return mp(s1,s2);
119
120 }
121
122 main(){
123     int i,j,k,a,b;
124     char o;
125     cin >> n;
126     st.resize(4*n);
127     A.resize(2*n);
128     st.assign(4*n,mp(-1,-1));
129     A.assign(2*n,-1);
130     for(i=0;i<n;i++){
131         cin >> A[i];
132     }
133     build(1,0,n-1);
134     cin >> m;
135     for(i=0;i<m;i++){
136         cin >> o >> a >> b;
137         if(o=='Q'){
138             ii aux = query(1,0,n-1,a-1,b-1);
139             cout << A[aux.F]+A[aux.S]  << endl;
140         }
```

```
141             else{
142                 update(1,0,n-1,a-1,b);
143             }
144         }
145 
146     }
147 
148 
149 
150 
151 
```