# 1754 I Hate It

**Problem Description**

很多学校流行一种比较的习惯。老师们很喜欢询问，从某某到某某当中，分数最高的是多少。  
这让很多学生很反感。  
不管你喜不喜欢，现在需要你做的是，就是按照老师的要求，写一个程序，模拟老师的询问。当然，老师有时候需要更新某位同学的成绩。

**Input**

本题目包含多组测试，请处理到文件结束。  
在每个测试的第一行，有两个正整数 N 和 M ( 0<N<=200000,0<M<5000 )，分别代表学生的数目和操作的数目。  
学生ID编号分别从1编到N。  
第二行包含N个整数，代表这N个学生的初始成绩，其中第i个数代表ID为i的学生的成绩。  
接下来有M行。每一行有一个字符 C (只取'Q'或'U') ，和两个正整数A，B。  
当C为'Q'的时候，表示这是一条询问操作，它询问ID从A到B(包括A,B)的学生当中，成绩最高的是多少。  
当C为'U'的时候，表示这是一条更新操作，要求把ID为A的学生的成绩更改为B。

**Output**

对于每一次询问操作，在一行里面输出最高成绩。

**Sample Input**

5 6

1 2 3 4 5

Q 1 5

U 3 6

Q 3 4

Q 4 5

U 2 9

Q 1 5

**Sample Output**

5

6

5

9

代码清单：

#include <iostream>

#include <stdio.h>

#include <cstring>

using namespace std;

struct node

{

int left, right, mid;

int maxn;

}w[4\*200005];

int val[200005];

int creat(int a, int b, int r) //建树

{

if(a == b)

{

w[r].left = w[r].right = a;

w[r].maxn = val[a];

return w[r].maxn;

}

w[r].left = a;

w[r].right = b;

w[r].mid = (a+b)/2;

w[r].maxn = max(creat(a, w[r].mid, 2\*r), creat(w[r].mid+1, b, 2\*r+1));

return w[r].maxn;

}

void update(int n, int m, int r) //更新

{

if(w[r].left == n && w[r].right == n)

{

w[r].maxn = m;

return;

}

if(n > w[r].mid)

{

update(n, m, 2\*r+1);

w[r].maxn = max(w[2\*r].maxn, w[2\*r+1].maxn);

}

else

{

update(n, m, 2\*r);

w[r].maxn = max(w[2\*r].maxn, w[2\*r+1].maxn);

}

}

int query(int a, int b, int r) //询问

{

if(w[r].left == a && w[r].right == b)

return w[r].maxn;

else if(a > w[r].mid)

return query(a, b, 2\*r+1);

else if(b <= w[r].mid)

return query(a, b, 2\*r);

else

return max(query(a, w[r].mid, 2\*r), query(w[r].mid+1, b, 2\*r+1));

}

int main()

{

char sh;

int i, t, n, x, y;

while(scanf("%d %d", &n, &t) != EOF)

{

for(i = 1; i <= n; i++)

{

scanf("%d", &val[i]);

}

creat(1, n, 1);

while(t--)

{

cin>>sh;

scanf("%d %d", &x, &y);

if(sh == 'Q')

printf("%d\n", query(x, y, 1));

else if(sh == 'U')

update(x, y, 1);

}

}

return 0;

}