

# [Vector]. Bài 2. Erase và insert

**Giới hạn thời gian:** 1.0s    **Giới hạn bộ nhớ:** 256M

Cho vector và 2 thao tác.

**Thao tác 1** : Chèn phần tử vào vị trí bất kì trong vector.

**Thao tác 2** : Xóa phần tử ở vị trí bất kì trong vector.

Đối với thao tác thứ 1, giả sử vector đang có **N** phần tử, chỉ số chèn hợp lệ sẽ là từ 0 tới **N**, ngoài ra các vị trí không hợp lệ sẽ không thực hiện chèn.

Tương tự đối với thao tác thứ 2, giả sử vector đang có **N** phần tử thì chỉ số xóa hợp lệ sẽ là từ 0 tới **N - 1**, nếu vị trí xóa không hợp lệ hoặc vector sẽ không thực hiện xóa.

Nhiệm vụ của bạn là thực hiện 1 loạt các thao tác này và in ra mảng sau khi thực hiện xong mọi thao tác. Nếu vector rỗng in ra **EMPTY**, ngược lại in ra các phần tử trong vector trên 1 dòng.

**Gợi ý :**

```
int m; cin >> m;
vector<int> v(m);
for(int i = 0; i < m; i++) cin >> v[i];
int n; cin >> n;
while(n--){
    int tt;
    cin >> tt;
    if(tt == 1){

    }
    else{

    }
}
// In ra vector
```

## Đầu vào

Dòng 1 là **M** : số lượng phần tử trong vector.

Dòng 2 là **M** số trong vector.

Dòng 3 là **N** : số thao tác được thực hiện. **N** dòng tiếp theo mỗi dòng mô tả thao tác, nếu thao tác là 1 sẽ có thêm vị trí chèn và giá trị được chèn, nếu thao tác là 2 sẽ có thêm vị trí xóa.

---

## Giới hạn

$1 \leq N \leq 200$

Các phần tử thêm vào vector là số nguyên int 32bit.

---

## Đầu ra

In ra **EMPTY** nếu vector rỗng, ngược lại in ra các phần tử trong vector trên 1 dòng.

---

## Ví dụ :

---

### Input 01

```
5
40 87 73 47 22
5
1 3 21
2 3
2 0
2 1
1 0 70
```

### Output 01

```
70 87 47 22
```