

C++-队列数组排序-幼儿园两个班的小朋友在排队时混在了一起

题目描述：

幼儿园两个班的小朋友在排队时混在了一起，每位小朋友都知道自己是否与前面一位小朋友是否同班，请你帮忙把同班的小朋友找出来。

小朋友的编号为整数，与前一位小朋友同班用 Y 表示，不同班用 N 表示。

输入描述：

输入为空格分开的小朋友编号和是否同班标志。

比如：6/N 2/Y 3/N 4/Y，表示共 4 位小朋友，2 和 6 同班，3 和 2 不同班，4 和 3 同班。

其中，小朋友总数不超过 999，每个小朋友编号大于 0，小于等于 999。

不考虑输入格式错误问题。

输出描述：

输出为两行，每一行记录一个班小朋友的编号，编号用空格分开。且：

- 1、编号需要按照大小升序排列，分班记录中第一个编号小的排在第一行。
- 2、若只有一个班的小朋友，第二行为空行。
- 3、若输入不符合要求，则直接输出字符串 ERROR。

补充说明：

示例 1

输入：

1/N 2/Y 3/N 4/Y

输出：

1 2

3 4

说明：

2 的同班标记为 Y，因此和 1 同班。

3 的同班标记为 N，因此和 1、2 不同班。

4 的同班标记为 Y，因此和 3 同班。

所以 1、2 同班，3、4 同班，输出为

1 2

3 4

```
#include <iostream>
```

```
#include <vector>
```

```
#include <algorithm>
```

```
#include <string>
```

```
using namespace std;
```

```
void getnumber(string& input,int& x, int& number,bool& ifsame,bool& error){
```

```
    int n=0;
```

```
    while(input[x]>='0'&&input[x]<='9'){
```

```
        n=n*10 + (input[x]-'0');
```

```

        x++;
    }
    number=n;
    if(input[x]!='/' || number<0 || number>999){
        error=true;
        return;
    }
    x++;
    if(input[x]=='Y'){
        ifsame=true;
    }else if(input[x]=='N'){
        ifsame=false;
    }else{
        error=true;
        return;
    }
    x++;
}

```

```

void output(vector<int>& A){
    for(int i=0;i<A.size();i++){
        if(i!=A.size()-1){
            cout<<A[i]<<" ";
        }else{
            cout<<A[i];
        }
    }
}

```

```

int main() {
    string input;
    getline(cin,input);

    vector<int> A,B;
    bool ifA=true;
    int number;
    bool ifsame;
    int x=0;
    bool error=false;

    while(x<input.size() && error==false){
        getnumber(input,x,number,ifsame,error);
        if(error==true){
            cout<<"ERROR"<<endl;

```

```

        return 0;
    }
    if(x==0){
        A.push_back(number);
    }else if(ifsame==true){
        if(ifA==true){
            A.push_back(number);
        }else{
            B.push_back(number);
        }
    }else{
        if(ifA==true){
            B.push_back(number);
            ifA=false;
        }else{
            A.push_back(number);
            ifA=true;
        }
    }
    x++;
}
sort(A.begin(),A.end());
sort(B.begin(),B.end());
if(B.size()==0){
    output(A);
    return 0;
}else if(A[0]<B[0]){
    output(A);
    cout<<endl;
    output(B);
}else{
    output(B);
    cout<<endl;
    output(A);
}
return 0;
}

```

// 64 位输出请用 printf("%lld")