//문제 1

function solution1() {

let a = gets().split(' ');

let b = gets().split(' ');

let cnt = 0;

for (let i = 0; i < a[0]; i++) {

for (let j = 1; j < a[0]; j++) {

if (a[2] == 'A') {

//정방향 스왑

if (b[j - 1] > b[j]) {

let temp = b[j - 1];

b[j - 1] = b[j];

b[j] = temp;

}

} else {

//역방향 스왑

if (b[j - 1] < b[j]) {

let temp = b[j - 1];

b[j - 1] = b[j];

b[j] = temp;

}

}

}

//카운팅 후 종료조건 제시

cnt++;

if (cnt == a[1]) break;

}

print(b.join(' '));

}

//문제 2

function solution2(){

let a = gets().split(' ');

let b = gets().split(' ');

let count = 0;

let ok = false;

let cnt = Number(a[1]);

function mergeSort(m){

if(m.length <= 1){

return m;

}

let mid = Math.floor(m.length / 2);

let left = m.slice(0, mid);

let right = m.slice(mid);

left = mergeSort(left);

right = mergeSort(right);

return merge(left, right);

function merge(left, right){

let result = [];

//종료 조건 : 정해진 횟수에 도달하면 그상태로 종료

if(count == cnt) {

return result = left.concat(right);

}

//병합정렬 과정 수행

while(left.length != 0 && right.length != 0){

if(a[2] == 'A'){

//정방향 정렬

if(left[0] <= right[0]){

result.push(left[0]);

left.shift();

}else{

result.push(right[0]);

right.shift();

}

}else{

//역방향 정렬

if(left[0] > right[0]){

result.push(left[0]);

left.shift();

}else{

result.push(right[0]);

right.shift();

}

}

}

while(left.length != 0){

result.push(left[0]);

left.shift();

}

//ok를 없애고 시도해보고 싶었으나 코딩테스트 사이트 기간 종료됨..

while(right.length != 0 && !ok){

result.push(right[0]);

right.shift();

}

//과정 진행 후 카운팅

count++;

return result;

}

}

//출력

print(mergeSort(b).join(' '));

}

//문제 3

function solution3() {

let a = gets().split(' ');

let leng = a.length;

let sorted = Array.from(Array(leng), () => Array(3).fill(0)); // 단어별로 에너그램화 되어있음

for (let i = 0; i < a.length; i++) {

//소문자화, 애너그램화 할 공간생성, 갯수 생성

sorted[i][0] = a[i].toLowerCase();

sorted[i][1] = '';

sorted[i][2] = 0;

}

for (let i = 0; i < a.length; i++) {

//애너그램 판별하기

sorted[i][1] = sorted[i][0].split('').sort().join('');

for (let j = 0; j <= i; j++) {

//기존에 동일한 애너그램 그룹이 있으면 그 그룹에 ++를 하고 마지막에 그 값을 추가한다.

if (sorted[i][1] == sorted[j][1]) {

sorted[j][2]++;

sorted[j].push(sorted[i][0]);

break;

}

}

}

//애너그램된 값들만 추출하기 위해 앞의 값 3개(애너그램 전, 애너그램후, 갯수)를 제거후 정렬

for (let i = 0; i < sorted.length; i++) {

sorted[i] = sorted[i].slice(3).sort();

}

//값이 0인 배열값 제거

sorted = sorted.filter((e) => e.length !== 0);

//문자열 순 정렬

sorted.sort((a, b) => {

return a > b ? 1 : -1;

});

//출력

for (let i of sorted) {

print(i.join(' '));

}

}