### (훈련반1) Level23.5

가지치기를 얼만큼 잘 하느냐에 따라서 재귀함수의 동작 속도는 더 빨라질 수 있습니다.

그리고 via 전역배열은 앞으로 배울 DFS 알고리즘에 활용되는 방법입니다.

Trace를 통해 개념을 확실하게 잡고 넘어가주시면 됩니다.

## Level23.5 왼쪽, 오른쪽 이동 [난이도 : 2]

**문제 1번** [[숙제 목록보기](http://quest.mincoding.co.kr/contest.php?cid=1065)]



위 배열을 하드코딩 해주세요.

**그리고 R 또는 L 문자 4개를 입력** 받습니다.

**R은 right 방향**을 의미하고

**L은 left 방향**을 의미 합니다.

아래 그림과 같이

**R을 입력** 받으면 숫자를 **오른쪽으로 한칸씩 모두 이동**시키는데 **맨 뒤에 있는 숫자는 맨앞**으로 와야합니다.



반대로 **L을 입력** 받으면 **숫자를 왼쪽으로 한칸씩 모두 이동** 시키고 **맨 앞에 있는 숫자는 맨 뒤**로 보냅니다.



**R 또는 L을 4번 입력 받은 후 처리된 결과를 출력** 해주세요.

## 입력 예제

R

R

R

L

## 출력 결과

9 7 3 5 1

**[소스 코드]**

#include <iostream>

#include <string>

#include <vector>

#include <list>

using std::cout;

using std::cin;

using std::endl;

using std::string;

using std::vector;

using std::list;

#define STRMAX 10

#define COL 4

#define ROW 5

#define MAX 5

int main()

{

int arr[MAX] = { 3,5,1,9,7 };

char input[MAX - 1] = {};

int index[10] = {};

for (int i = 0; i < MAX - 1; i++)

{

cin >> input[i];

}

for (int i = 0; i < MAX; i++)

{

index[arr[i]] = i;

}

for (int j = 0; j < MAX - 1; j++)

{

switch (input[j])

{

case 'R':

for (int i = 0; i < MAX; i++)

{

index[arr[i]] += 1;

if (index[arr[i]] == MAX)

index[arr[i]] = 0;

}

break;

case 'L':

for (int i = 0; i < MAX; i++)

{

index[arr[i]]-=1;

if (index[arr[i]] == -1)

index[arr[i]] = MAX-1;

}

break;

}

}

int temp[MAX] = {};

for (int i = 0; i < MAX; i++)

{

int idx = index[arr[i]];

temp[idx] = arr[i];

}

for (int i = 0; i < MAX; i++)

{

cout<<temp[i]<<' ';

}

return 0;

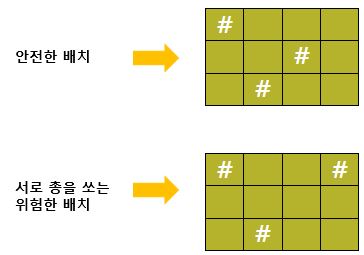
}

## Level23.5 암살자 존휙 [난이도 : 3]

**문제 2번** [[숙제 목록보기](http://quest.mincoding.co.kr/contest.php?cid=1065)]

**#은 암살자들이 있는 위치** 입니다. **3명의 암살자의 위치를 입력** 받으세요.

만약, 직선거리에 상대방이 있다면 서로 총을 쏘게 됩니다.



**세명의 좌표를 입력** 받고

서로 총을 쏘지 않는 안전한 위치라면 **"안전"** 출력

그렇지않다면 **"위험"**을 출력 해주세요.

## 입력 예제

0 0

1 2

2 1

## 출력 결과

안전

**[소스 코드]**

#include <iostream>

#include <string>

#include <vector>

#include <list>

using std::cout;

using std::cin;

using std::endl;

using std::string;

using std::vector;

using std::list;

#define STRMAX 10

#define COL 3

#define ROW 4

#define MAX 5

struct Index

{

int x;

int y;

};

int main()

{

int col[COL] = {};

int row[ROW] = {};

Index index[COL];

for (int i = 0; i < COL; i++)

{

cin >> index[i].y >> index[i].x;

col[index[i].y]++;

row[index[i].x]++;

}

bool flag = false;

for (int i = 0; i < COL; i++)

{

if (col[i] >= 2)

flag = true;

}

for (int i = 0; i < ROW; i++)

{

if (row[i] >= 2)

flag = true;

}

if (flag == true)

cout << "위험";

else

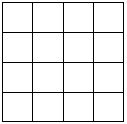
cout << "안전";

return 0;

}

## Level23.5 네모네모 더하기 [난이도 : 3]

**문제 3번** [[숙제 목록보기](http://quest.mincoding.co.kr/contest.php?cid=1065)]



**4x4 배열을 만들고 (0,0)~(2,2)까지 3 x 3칸에 다가 아홉 숫자를 입력** 받으세요.

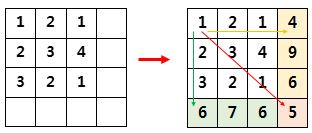
예를들어

**1 2 1**

**2 3 4**

**3 2 1**  을 입력 받았다면 아래와 같이 배열값이 넣어지게 됩니다.

그리고 빈칸에는 **가로줄의 합& 세로줄의 합 & 대간선줄의 합이 계산**되어 채워 집니다.



배열에 모든 값이 채워지면 출력합니다.

적절한 for문을 사용하여 이 프로그램을 만들어 주세요

## 입력 예제

1 2 1

2 3 4

3 2 1

## 출력 결과

1 2 1 4

2 3 4 9

3 2 1 6

6 7 6 5

**[소스 코드]**

#include <iostream>

#include <string>

#include <vector>

#include <list>

using std::cout;

using std::cin;

using std::endl;

using std::string;

using std::vector;

using std::list;

#define STRMAX 10

#define COL 3

#define ROW 4

#define MAX 4

int main()

{

int arr[MAX][MAX] = {};

for (int i = 0; i < 3; i++)

{

for (int j = 0; j < 3; j++)

{

cin >> arr[i][j];

}

}

int sum = 0;

int index = 0;

for (int i = 0; i < 3; i++)

{

sum = 0;

for (int j = 0; j < MAX; j++)

{

if (j == MAX - 1)

arr[i][j] = sum;

else

sum += arr[i][j];

}

}

sum = 0;

for (int i = 0; i < MAX; i++)

{

sum = 0;

for (int j = 0; j < MAX; j++)

{

if (j == MAX - 1)

arr[j][i] = sum;

else

sum += arr[j][i];

}

}

sum = 0;

for (int i = 0; i < MAX; i++)

{

for (int j = 0; j < MAX; j++)

{

if (i == j && i == MAX - 1)

arr[i][j] = sum;

else if (i == j)

sum += arr[i][j];

}

}

for (int i = 0; i < MAX; i++)

{

for (int j = 0; j < MAX; j++)

{

cout << arr[i][j]<<' ';

}

cout << endl;

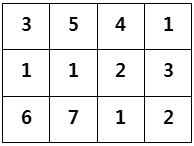
}

return 0;

}

## Level23.5 숫자 transformation [난이도 : 4]

**문제 4번** [[숙제 목록보기](http://quest.mincoding.co.kr/contest.php?cid=1065)]



위 숫자들을 하드코딩 해주세요.

그리고 각기 다른 숫자 4개를 배열에 입력 받으세요.

**예로들어 1 3 7 6 을 입력 받았다고 한다면, 이차배열의 값을**

숫자 **1**을 **3**으로 변경

숫자 **3**을 **7**로 변경

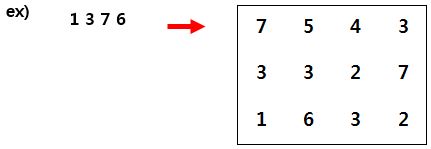
숫자 **7**을 **6**으로 변경

숫자 **6**을 **1**로 변경

**하시면 됩니다.**

**(이 외 나머지 숫자는 그대로 두시면 됩니다)**

**변경된 이차배열 값을 출력해주세요.**



## 입력 예제

1 3 7 6

## 출력 결과

7 5 4 3

3 3 2 7

1 6 3 2

**[소스 코드]**

#include <iostream>

#include <string>

#include <vector>

#include <list>

using std::cout;

using std::cin;

using std::endl;

using std::string;

using std::vector;

using std::list;

#define STRMAX 10

#define COL 3

#define ROW 4

#define MAX 4

int main()

{

int arr[COL][ROW] =

{ 3,5,4,1,

1,1,2,3,

6,7,1,2 };

int input[ROW] = {};

for (int i = 0; i < ROW; i++)

{

cin >> input[i];

}

for (int j = 0; j < COL; j++)

{

for (int z = 0; z < ROW; z++)

{

for (int i = 0; i < ROW; i++)

{

if (input[i] == arr[j][z])

{

if (i + 1 >= ROW)

{

arr[j][z] = input[0];

}

else

arr[j][z] = input[i + 1];

break;

}

}

}

}

for (int i = 0; i < COL; i++)

{

for (int j = 0; j < ROW; j++)

{

cout << arr[i][j]<<' ';

}

cout << endl;

}

return 0;

}

## Level23.5 자기자리 찾기 [난이도 : 6]

**문제 5번** [[숙제 목록보기](http://quest.mincoding.co.kr/contest.php?cid=1065)]

**8개의 숫자를 배열에 입력받아주세요.**

**배열에서 가장 왼쪽에 있는 숫자를 "피벗"**이라고 합니다.

만약 아래와 같이 4 1 7 9 6 3 3 6을 입력받으면 피벗은 4가 됩니다.



이 배열을 가지고 아래에 나와있는 규칙대로 숫자들을 옮기다보면,

신기하게도

**피벗 왼쪽에는 피벗보다 작은숫자**들이

**피벗 오른쪽에는 피벗보다 큰 숫자**들로 구성됩니다.

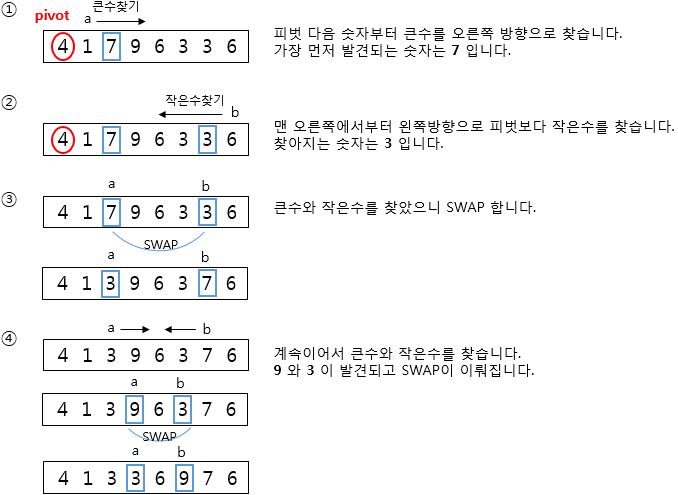


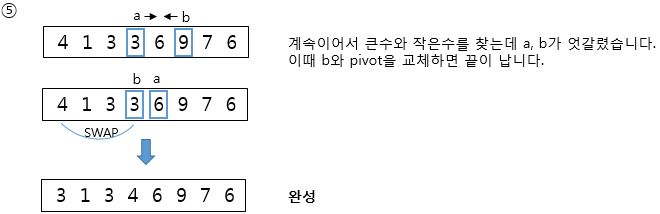
**아래의 규칙에 따라 숫자를 옮기고, 결과를 출력 해 주세요**

**요약 :**

**a는 pivot보다 큰수를 찾아야하고**

**b는 pivot보다 작은수를 찾아서 SWAP 해야합니다.**





## 입력 예제

4 1 7 9 6 3 3 6

## 출력 결과

3 1 3 4 6 9 7 6

**[소스 코드]**

## Level23.5 황금좌표 찾기 [난이도 : 5]

**문제 6번** [[숙제 목록보기](http://quest.mincoding.co.kr/contest.php?cid=1065)]

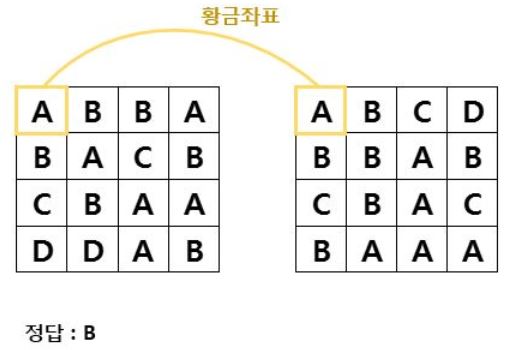
아래 그림을 보면 두개의 4x4 배열이 있습니다.

**왼쪽배열 (4x4)는 입력받고,**

**오른쪽배열 (4x4)는 하드코딩 해 주세요.**

이 **두 배열에서 같은 좌표값이 같은 알파벳을 가지고 있으면** **황금 좌표** 입니다.

**황금 좌표를 가장 많이 가진 알파벳을 찾아 출력 해주세요.**



위 예제에서는 B가 황금좌표를 4개를 가지고 있기 때문에

정답은 B입니다.

## 입력 예제

ABBA

BACB

CBAA

DDAB

## 출력 결과

B

**[소스 코드]**