

## (훈련반1) Level22

재귀호출을 하다 보면 어느 시점에 있는지 정확히 알기 힘듭니다.

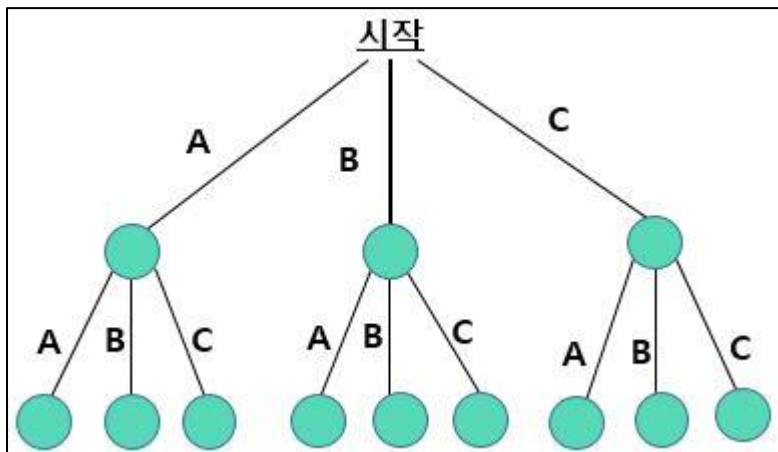
path전역배열을 이용하면 경로를 알 수 있습니다!

재귀호출에서 필수로 알아야 하는 path를 이해해봅시다.

## Level22 경로를 출력하자 [난이도 : 2]

문제 1번 [\[숙제 목록보기\]](#)

Level2에 도달했을 때마다, path를 모두 출력 해주세요.



## 출력 결과

AA  
AB  
AC  
BA  
BB  
BC  
CA  
CB  
CC

```
#include <iostream>
using namespace std;

const int maxLevel = 2;
const int route = 3;
char arr[5] = {};

void recursive(int level)
{
    if (level == maxLevel)
    {
        cout << arr << endl;
        return;
    }
    for (int i = 0; i < route; ++i)
    {
        arr[level] = 'A' + i;
        recursive(level + 1);
        arr[level] = '\0';
    }
}

int main(void)
{
    int startLevel = 0;
    recursive(startLevel);

    return 0;
}
```

# Level22 같은문장 찾기 [난이도 : 1]

문제 2번 [[숙제](#) [목록보기](#)]

<cstring> strcmp를 사용해서 문제를 풀어주세요.

세문장을 입력 받으세요. (최대글자는 10글자)

세문장이 모두 같으면 "WOW"

두문장이 같으면 "GOOD"

모두 다르면 "BAD" 를 출력 해주세요.

## 입력 예제

abcd

abcd

abcd

## 출력 결과

WOW

```
#include <iostream>
using namespace std;

const int strCnt = 3;

int main(void)
{
    char str[strCnt][5] = {};

    for (int i = 0; i < strCnt; ++i)
        cin >> str[i];

    if (strcmp(str[0], str[1]) == 0 && strcmp(str[0], str[2]) == 0)
        cout << "WOW";
    else if (strcmp(str[0], str[1]) == 0 || strcmp(str[0], str[2]) == 0 || strcmp(str[1],
str[2]) == 0)
        cout << "GOOD";
    else
        cout << "BAD";

    return 0;
}
```

# Level22 경로 이름은 BGTK [난이도 : 3]

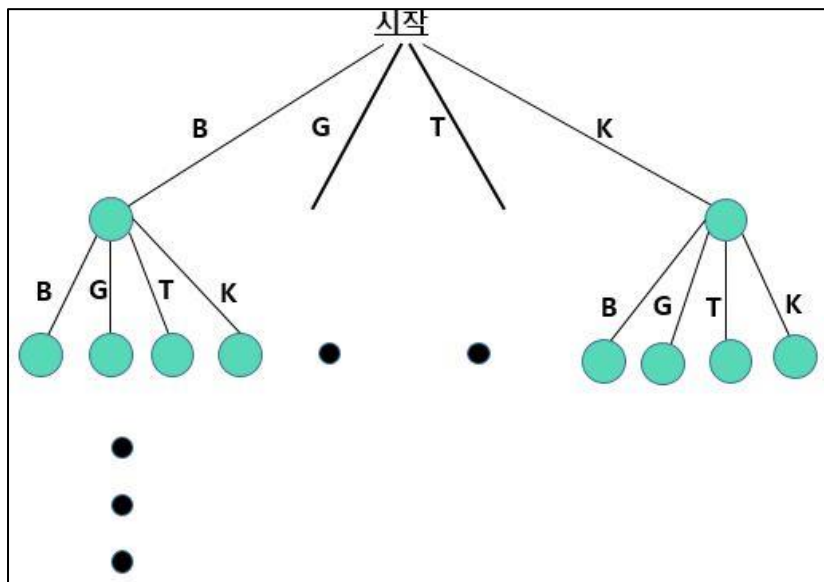
문제 3번 [숙제 목록보기]

다음과 같은 구조로 Level을 입력받고

입력받은 Level에 도달했을 때

path를 출력 해주세요.

(2 <= LEVEL <= 4)



## 입력 예제

2

## 출력 결과

BB

BG

BT

BK

GB

GG

GT

GK

TB

TG

TT

TK

KB

KG

KT

KK

```
#include <iostream>
using namespace std;

int maxLevel = 0;
const int startLevel = 0;
const int routeCnt = 4;
char path[4] = {};          // 2 <= level <= 4

char choiceRoute(int num)
{
    char arrRoute[routeCnt] = {'B', 'G', 'T', 'K'};

    return arrRoute[num];
}

void recursive(int level)
{
    if (level == maxLevel)
    {
        cout << path << endl;
        return;
    }
    for (int i = 0; i < routeCnt; ++i)
    {
        path[level] = choiceRoute(i);
        recursive(level + 1);
        path[level] = '\\0';
    }
}

int main(void)
{
    cin >> maxLevel;

    recursive(startLevel);

    return 0;
}
```

# Level22 Up & Down [난이도 : 2]

문제 4번 [숙제 [목록보기](#)]

up 또는 down 문장을 5개 입력 받으세요.

주인공이 1층에 서있는데

up이 입력으로 들어오면 2층으로 가고 (up 입력시 한층 올라감)

down이 들어오면 B1층으로 갑니다. (down 입력시 한층 내려감)

up/down 을 5회 입력 받았을 때 주인공의 위치를 출력 하세요.

ex)

입력:

up

down

down

down

down

출력: B3

## 입력 예제

up

down

down

down

down

## 출력 결과

B3



```
#include <iostream>
using namespace std;

const int commandCount = 5;
const int maxLength = 5;

int main(void)
{
    int position = 1;
    // 명령 입력받기
    char command[commandCount][maxLength] = {};
    for (int i = 0; i < commandCount; ++i)
        cin >> command[i];

    // 층수 이동
    for (int k = 0; k < commandCount; ++k)
    {
        if (strcmp(command[k], "up") == 0)
            ++position;
        else
            --position;
    }

    // 층 수 출력
    if (position > 0)
        cout << position;
    else
    {
        position *= (-1);
        position += 1;
        cout << 'B' << position;
    }
    return 0;
}
```

# Level22 청소당번 [난이도 : 4]

문제 5번 [숙제 목록보기]



1 ~ n 명의 친구가 4일치 청소당번 순서를 정하려고 합니다.

첫번째 친구가 4일치 모두 청소할 수도 있고, 다양한 경우가 발생합니다.

ex) n = 3명일때 가능한 청소스케줄

1111 / 1112 / 1113 / 1121 / 1122 / ... / 3332 / 3333

n명에 대해, 4일 동안의 청소스케줄을 모두 출력하는 프로그램을 작성해주세요.

## 입력 예제

2

## 출력 결과

1111

1112

1121

1122

1211

1212

1221

1222

2111

2112

2121

2122

2211

2212

2221

2222

```
#include <iostream>
using namespace std;

int people = 0;
const int period = 4;
char schedule[4] = {};

void recursive(int day)
{
    if (day == period)
    {
        cout << schedule << endl;
        return;
    }
    for (int i = 0; i < people; ++i)
    {
        schedule[day] = '1' + i;
        recursive(day + 1);
        schedule[day] = '\0';
    }
}

int main(void)
{
    cin >> people;

    int startDay = 0;
    recursive(startDay);

    return 0;
}
```

# Level22 가장 긴문장, 짧은 문장은 어디에

## [난이도 : 3]

문제 6번 [[숙제](#) [목록보기](#)]

string 1차원 배열에 네문장을 입력 받으세요.

가장 긴 문장과 가장 짧은 문장이 몇번째 index에 있는 문장인지 index를 출력 해주세요

ex)

[입력]

apple

microsoft

google

xiaomi

[출력]

긴문장:1 //(microsoft)

짧은문장:0 //(apple)

[hint]

문장의 길이를 구하는 방법

```
string a = "mincoding";
```

```
int len = a.length();
```

## 입력 예제

apple

microsoft

google

xiaomi

# 출력 결과

긴문장:1

짧은문장:0

```
#include <iostream>
using namespace std;

const int inputCnt = 4;

int main(void)
{
    char str[inputCnt][20] = {};
    for (int i = 0; i < inputCnt; ++i)
        cin >> str[i];

    int arrLength[inputCnt] = {};
    for (int k = 0; k < inputCnt; ++k)
    {
        arrLength[k] = (int)strlen(str[k]);
    }
    int maxIdx = 0;
    int minIdx = 0;

    for (int o = 0; o < inputCnt; ++o)
    {
        if (arrLength[o] > arrLength[maxIdx])
            maxIdx = o;

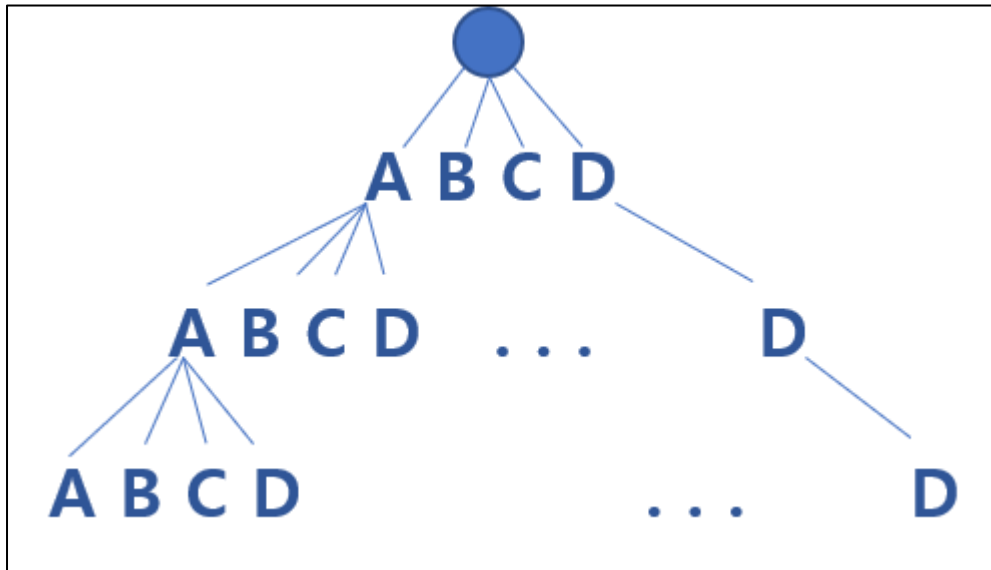
        if (arrLength[o] < arrLength[minIdx])
            minIdx = o;
    }

    cout << "긴문장:" << maxIdx << " // (" << str[maxIdx] << ")\n";
    cout << "짧은문장:" << minIdx << " // (" << str[minIdx] << ")\n";

    return 0;
}
```

## Level22 몇번째 행운 [난이도 : 5]

문제 7번 [숙제 [목록보기](#)]



위와 같은 트리모양으로 재귀호출 프로그램을 만드려 합니다.

Level 3에 도착했을때 입력 받은 PATH가 몇번째로 만들어지는지 출력 해주세요.

ex)

AAA => 1번째

### 입력 예제

AAD

### 출력 결과

4번째

```

#include <iostream>
using namespace std;

const int route = 4;
char path[10] = {};
char input[4] = {};
int pathNum = 0;
bool isFind = false;

void recursive(int level)
{
    if (level == 3)
    {
        ++pathNum;
        if (strcmp(path, input) == 0)
        {
            cout << pathNum << "번째";
            isFind = true;
            //return;
        }
        return;          // *반환은 내부 조건 밖에 있어야 한다.
    }
    for (int i = 0; i < route; ++i)
    {
        path[level] = 'A' + i;
        recursive(level + 1);
        path[level] = '\0';
        if (isFind)
            break;
    }
}

int main(void)
{
    cin >> input;
    int initLevel = 0;
    recursive(initLevel);

    return 0;
}

```

## Level22 3차 배열에서 MAX MIN 찾기

[난이도 : 3]

문제 8번 [[숙제](#) [목록보기](#)]

아래와 같이 3차 배열을 하드코딩 해 주세요

2	4	2	3	7	3
1	5	3	6	1	5
[0]		[1]		[2]	

숫자 한개를 입력 받고, 입력받은 위치의 MAX와 MIN값을 출력 해 주세요

ex) 만약 0을 입력받았다면 [0]에 해당하는 곳에서 MAX값과 MIN값을 출력 하면 됩니다

MAX=5

MIN=1

### 입력 예제

1

### 출력 결과

MAX=6

MIN=2



```

#include <iostream>
using namespace std;

const int inputCnt = 5;
const int yLength = 2;
const int xLength = 2;

struct Point
{
    Point()
        :val(0)
        , posX(0)
        , posY(0)
    {
    }
    int val;
    int posX;
    int posY;
};

int main(void)
{
    int arr3D[3][2][2] =
    {
        {
            2,4,
            1,5
        },
        {
            2,3,
            3,6
        },
        {
            7,3,
            1,5
        }
    };
    int input = 0;
    cin >> input;

    Point max;
    max.val = arr3D[input][0][0];
    Point min;
    min.val = arr3D[input][0][0];
}

```

```
for (int y = 0; y < yLength; ++y)
{
    for (int x = 0; x < xLength; ++x)
    {
        if (max.val < arr3D[input][y][x])
        {
            max.val = arr3D[input][y][x];
            max.posY = y;
            max.posX = x;
        }

        if (min.val > arr3D[input][y][x])
        {
            min.val = arr3D[input][y][x];
            min.posY = y;
            min.posX = x;
        }
    }
}

cout << "MAX=" << max.val << endl;
cout << "MIN=" << min.val;

return 0;
}
```

# Level22 암호 해독하기 [난이도 : 3]

문제 9번 [[숙제](#) [목록보기](#)]

암호표에 암호문들이 있습니다.

Jason
Dr.tom
EXEXI
GK12P
POW

암호문을 입력 받아주세요.

암호가 맞으면 "암호해제" 라고 출력,

암호가 틀리면 "암호틀림" 라고 출력 해주세요.

## 입력 예제

abc

## 출력 결과

암호틀림

```
#include <iostream>
using namespace std;

const int inputCnt = 5;

int main(void)
{
    char secret[inputCnt][10] =
    {
        "Jason",
        "Dr.tom",
        "EXEXI",
        "GK12P",
        "POW"
    };

    char input[10] = {};
    cin >> input;

    bool isCorrect = false;
    for (int k = 0; k < inputCnt; ++k)
    {
        if (strcmp(input, secret[k]) == 0)
        {
            isCorrect = true;
            break;
        }
    }

    if (isCorrect)
        cout << "암호해제";
    else
        cout << "암호틀림";

    return 0;
}
```