

그 동안 문장을 다루는데 큰 불편함을 느꼈을 것입니다.

non-library 단계에서 library를 사용하는 방향으로 넘어가는 시점입니다.

cstring library를 사용해서 편리하게 문장을 다뤄 봅시다.

그리고 재귀호출 두번째 수업인 Branch와 Level에 대해 이해하고, 훈련 해 봅시다.

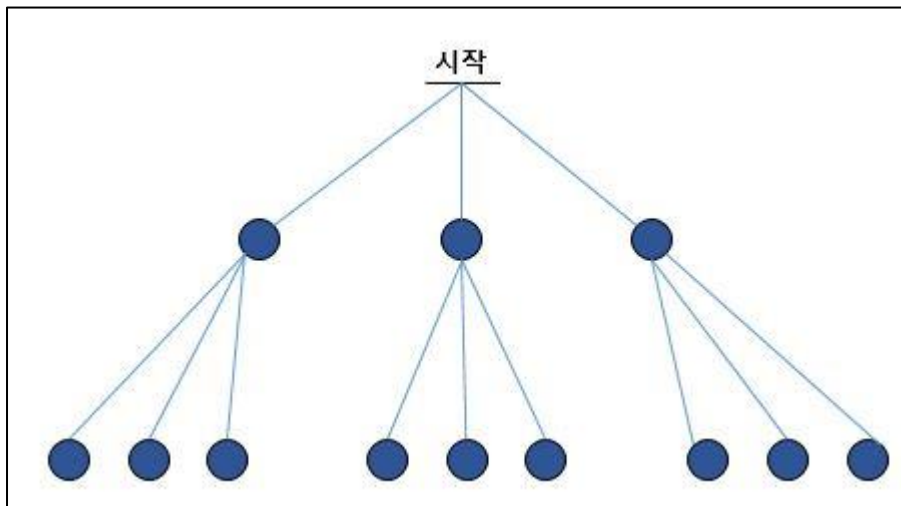
재귀호출은 어렵지 않습니다.

## Level21 재귀호출이 3개일때 [난이도 : 2]

문제 1번 [숙제 목록보기]

그림과 같이 동작되는 프로그램을 작성하세요.

(입출력 값이 없는 문제입니다, 소스코드만 작성 후 제출 해 주세요)



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

void func(int num)
{
    if (num == 2)
        return;
    func(num + 1);
    func(num + 1);
    func(num + 1);
}

int main(void)
{
    int num = 0;
    func(0);

    return 0;
}
```

# Level21 로그인 처리하기 [난이도 : 2]

문제 2번 [[숙제 목록보기](#)]

운영자의 ID와 PASSWORD는

"qlqlaqkq"

"tkaruqtkf" 입니다. (비빔밥 / 삼겹살)

ID와 PASSWORD를 입력받고,

운영자 ID와 PASSWORD가 정확히 일치한다면, "LOGIN"을 출력 해 주세요

아니면 "INVALID"를 출력 해 주세요

([cstring library](#)를 사용 해 주세요)

## 입력 예제

```
qlqlaqkq
```

```
tkaruqtkf
```

## 출력 결과

```
LOGIN
```

```

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

class User
{
public:
    User(string input_id, string input_pw)
        : db_id(input_id)
        , db_pw(input_pw)
    {
    }

    char isLoginable(string try_id, string try_pw)
    {
        if (db_id.compare(try_id) != 0)
            return 'I';    // 아이디 갯춘
        if (db_pw.compare(try_pw) != 0)
            return 'P';    // 비번 갯춘
        return 'A';        // 통과
    }

private:
    string db_id;    // 생성할때 만든 id
    string db_pw;    // 생성할때 만든 pw
};

int main(void)
{
    User admin("qlqlaqkq", "tkaruqtkf");

    string id, pw;
    cin >> id >> pw;

    char result = admin.isLoginable(id, pw);
    switch (result)
    {
        case 'A' :
            cout << "LOGIN";
            break;
        default :
            cout << "INVALID";
    }

    return 0;
}

```

# Level21 Level과 Branch [난이도 : 3]

문제 3번 [[숙제 목록보기](#)]

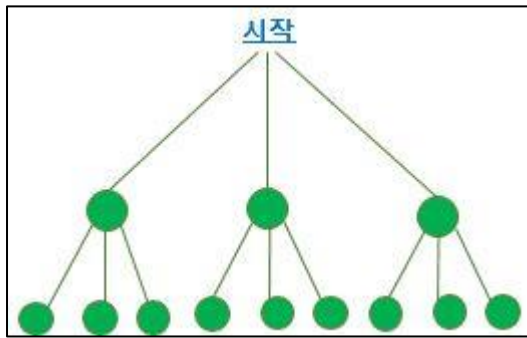
level과 branch를 입력 받고 그만큼 재귀호출을 해 주세요

(출력결과 없음)

ex)

level = 2

branch = 3



## 입력 예제

2

3

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

int level;
int branch;

void func(int initLevel)
{
    if (initLevel == level)
        return;
    for (int i = 0; i < branch; ++i)
        func(initLevel + 1);
}

int main(void)
{
    cin >> level >> branch;
    int initLevel = 0;
    func(initLevel);

    return 0;
}
```

# Level21 입력받은 Level까지 재귀함수 동작

## [난이도 : 3]

문제 4번 [[숙제](#) [목록보기](#)]

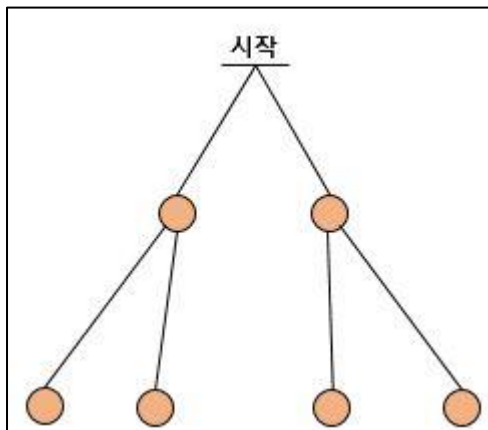
숫자 1개를 입력받고,

입력받은 Level 까지 재귀함수가 동작되도록 코딩해주세요

함수가 시작될 때 Level을 출력하면 됩니다.

ex1)

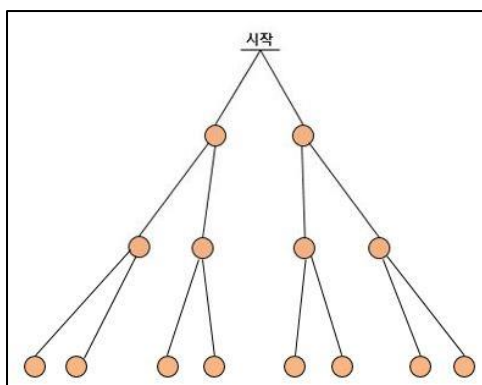
입력 : 2



출력결과 : 0122122

ex2)

입력 : 3



출력결과 : 012332331233233

# 입력 예제

2

# 출력 결과

0122122

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

int level = 0;
int branch = 2;

void func(int initLevel)
{
    cout << initLevel;
    if (initLevel == level)
        return;
    for (int i = 0; i < branch; ++i)
        func(initLevel + 1);
}

int main(void)
{
    cin >> level;
    int initLevel = 0;
    func(initLevel);

    return 0;
}
```



# Level21 긴문장을 맨 앞으로 [난이도 : 4]

문제 5번 [[숙제](#) [목록보기](#)]

세 문장을 2차원 배열에 입력 받으세요.(최대 10글자)

가장 긴문장과 첫 번째로 입력 받은 문장을 SWAP 한 후,

배열에 있는 문장을 모두 출력 해주세요.

(strlen과 strcpy를 사용하면 됩니다)

## 입력 예제

KFC

ABCDEFGF

BBQ

## 출력 결과

ABCDEFGF

KFC

BBQ

```

#include <iostream>
using namespace std;

const int lineCount = 3;
const int charCount = 10;

int main(void)
{
    // 문자열 입력
    char input[lineCount][charCount] = {};
    for (int i = 0; i < lineCount; ++i)
        cin >> input[i];

    // 문자열 길이구하기
    int arrLen[lineCount] = {}; // 문자열 길이 저장하는 배열
    int maxLen = -1;
    int maxIdx = -1;
    for (int k = 0; k < lineCount; ++k)
    {
        arrLen[k] = (int)strlen(input[k]);
        if (maxLen < arrLen[k])
        {
            maxLen = arrLen[k];
            maxIdx = k;
        }
    }

    if (maxIdx != -1 && lineCount > 0)
    {
        if(maxIdx != 0)
        {
            char temp[charCount] = {};
            strcpy_s(temp, input[maxIdx]);
            strcpy_s(input[maxIdx], input[0]);
            strcpy_s(input[0], temp);
        }
        for (int o = 0; o < lineCount; ++o)
        {
            cout << input[o] << endl;
        }
    }

    return 0;
}

```

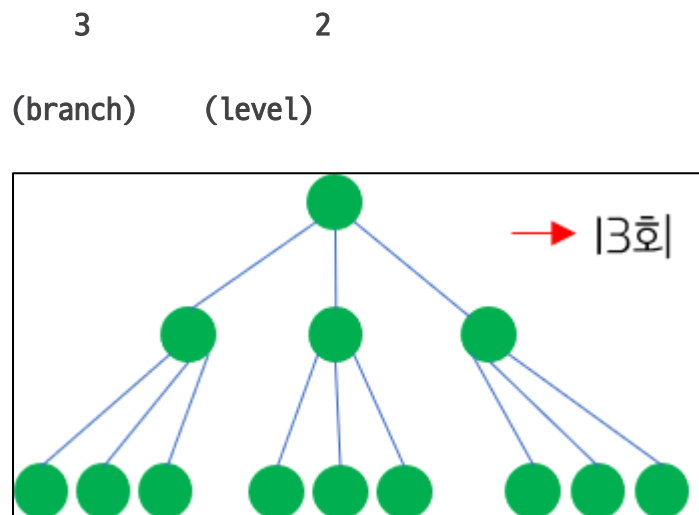
# Level21 재귀는 몇번 [난이도 : 5]

문제 6번 [숙제 [목록보기](#)]

branch와 Level을 입력받고 재귀 함수가 호출되는 횟수를 counting해서 출력해주세요.

예로들어 3 2 을 입력받으면 재귀함수는 총 13회 호출 됩니다.

ex)



[힌트]

전역변수를 두고 함수를 호출할 때마다 counting 해주면 됩니다.

## 입력 예제

4 3

## 출력 결과

85

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

int level = 0;
int branch = 0;
int cnt = 0;

void func(int initLevel)
{
    cnt++;
    if (initLevel == level)
        return;
    for (int i = 0; i < branch; ++i)
        func(initLevel + 1);
}

int main(void)
{
    cin >> branch >> level;
    int initLevel = 0;
    func(initLevel);
    cout << cnt;

    return 0;
}
```

# Level21 글자수만큼 손가락 접기

[난이도 : 3]

문제 7번 [[숙제](#) [목록보기](#)]

한문장을 입력받고 글자수를 구해주세요.

그리고 **재귀호출을 이용**해 다음과 같이 출력 해주세요.

만약 ABCDE를 입력받았다면 5글자 이므로 5 4 3 2 1 2 3 4 5 출력

만약 BBQ를 입력받았다면 3글자 이므로 3 2 1 2 3 을 출력 하면 됩니다.

(문장의 내용은 중요하지 않습니다. 글자수가 중요합니다.)

## 입력 예제

ABCDE

## 출력 결과

5 4 3 2 1 2 3 4 5

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

void func(size_t initLevel)
{
    cout << initLevel;
    if (initLevel == 1)
        return;
    func(initLevel - 1);
    cout << initLevel;
}

int main(void)
{
    string input = "";    // string이 내부에서 알아서 동적할당 해주고 해제도 해준다.
    cin >> input;

    size_t len = input.length();
    func(len);

    return 0;
}
```

# Level21 생일선물 마우스 [난이도 : 4]

문제 8번 [숙제 목록보기]

생일선물로 마우스를 직접 만들고,

내 연인에게 선물을 주며 고백하려 합니다.

마우스는 현재  $(y,x)=(5,5)$ 에 위치합니다.

up / down / left / right / click 명령어들이 존재합니다.

명령어 수(n)와 명령어들을 입력받고 명령어 대로 수행 해주세요.

## [명령어]

**up** : y축으로 한칸 -1

**down** : y축으로 한칸 +1

**left** : x축으로 한칸 -1

**right** : x축으로 한칸 +1

**click** : 현재 좌표 출력

## [입력예제]

4 //명령어의 수

up

click //출력

right

click //출력

## [출력결과]

4,5

4,6

## 입력 예제

3

down

down

click

## 출력 결과

7,5



```

#include <iostream>
using namespace std;

void func(int* x, int* y, string input)
{
    if (strcmp(input.c_str(), "up") == 0)
        --(*y);
    else if (strcmp(input.c_str(), "down") == 0)
        ++(*y);
    else if (strcmp(input.c_str(), "left") == 0)
        --(*x);
    else if (strcmp(input.c_str(), "right") == 0)
        ++(*x);
    else if (strcmp(input.c_str(), "click") == 0)
        cout << *y << ", " << *x << endl;
    else
        __noop;
}

int main(void)
{
    int mousePosX = 5;
    int mousePosY = 5;

    int cnt = 0;
    cin >> cnt;

    string* cmd = new string[cnt];
    for (int i = 0; i < cnt; ++i)
    {
        cin >> cmd[i];
    }

    for (int k = 0; k < cnt; ++k)
    {
        func(&mousePosX, &mousePosY, cmd[k]);
    }

    delete[] cmd;    // 배열
    cmd = nullptr;

    return 0;
}

```