

그 동안 for문을 활용한 다양한 코딩 기법을 배워보았고,

이제는 재귀호출을 활용한 코딩 기법을 시작하게 됩니다.

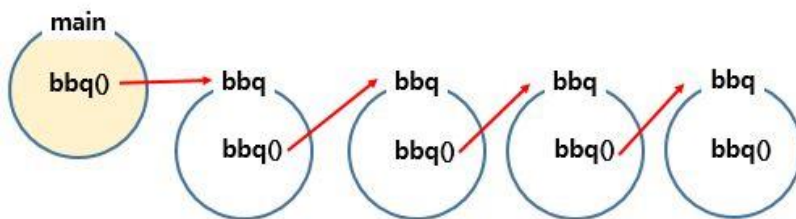
재귀호출은 처음 이해하기는 힘들지만, 익숙하게 되면 그리 어렵지 않습니다.

## Level20 무한 재귀 막기 [난이도 : 3]

문제 1번 [숙제 [목록보기](#)]

아래 그림과 같이 재귀 함수를 구현해주세요.

(전역변수를 쓰지 않습니다)



입출력 값이 없는 문제입니다.

Trace 연습을 많이 해야합니다.

F10, F11, ctrl + F10 버튼을 이용해서

능숙해지도록 연습을 꼭 해보세요!!

```
#include <iostream>
using namespace std;
void bbq(int num)
{
    if (num == 4)
        return;
    bbq(num + 1);
}
int main(void)
{
    int cnt = 1;
    bbq(cnt);
    return 0;
}
```

# Level20 번지점프 [난이도 : 3]

문제 2번 [[숙제](#) [목록보기](#)]

숫자 n을 입력 받으세요.

숫자 n부터 0까지 Count down 했다가

다시 돌아오는 수를 출력 하시면 됩니다.

ex) 4

4 3 2 1 0 1 2 3 4

ex ) 6

6 5 4 3 2 1 0 1 2 3 4 5 6

## 입력 예제

4

## 출력 결과

4 3 2 1 0 1 2 3 4

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

void func(int n)
{
    cout << n << " ";
    if (n == 0)
        return;
    func(n-1);
    cout << n << " ";
}

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

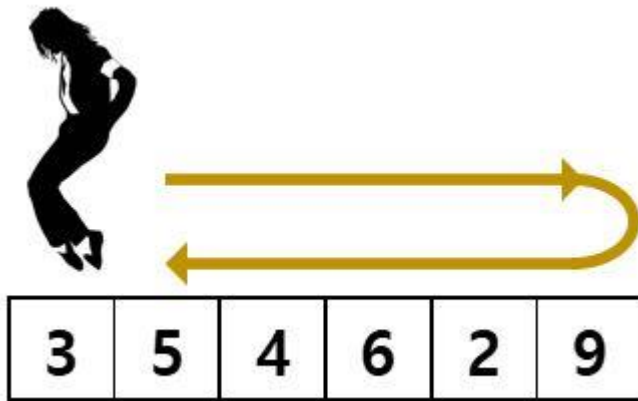
    return 0;
}
```

# Level20 마이클잭슨 무브먼트 [난이도 : 5]

문제 3번 [[숙제](#) [목록보기](#)]

마이클잭슨은 앞으로 갔다가 뒤로가는 백스텝 춤을 추곤합니다.

이 무브먼트(움직임)를 **재귀**를 사용해 출력 해주세요.



만약,

3 5 4 6 2 9    를 입력 받으면

3 5 4 6 2 9 2 6 4 5 3    이 출력 됩니다.

## 입력 예제

1 2 3 4 5 6

## 출력 결과

1 2 3 4 5 6 5 4 3 2 1

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

void func(int* arr, int idx)
{
    cout << arr[idx] << " ";
    if (idx == 5)
        return;
    func(arr, idx + 1);
    cout << arr[idx] << " ";
}

int main(void)
{
    int arr[6] = {};
    for (int i = 0; i < 6; ++i)
        cin >> arr[i];

    func(arr, 0);
    return 0;
}
```

# Level20 두칸씩 점프하기 [난이도 : 4]

문제 4번 [\[숙제 목록보기\]](#)

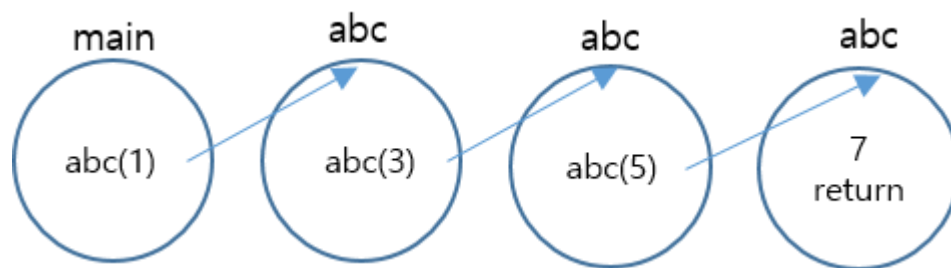
다음과 같이 동작하는 프로그램을 작성해 주세요

한번 재귀호출 될 때 마다, 2씩 증가됩니다.

그리고 3번 재귀 함수 진입 후, 리턴되면서 값을 출력 하면 됩니다.

만약 1을 입력받았다면,

출력 결과 : 7 5 3 1



## 입력 예제

1

## 출력 결과

7 5 3 1

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

void abc(int num)
{
    if (num > 7)
        return;
    abc(num + 2);
    cout << num << " ";
}

int main(void)
{
    abc(1);

    return 0;
}
```

# Level20 다섯글자 순차/역순으로 출력

문제 5번 [[숙제](#) [목록보기](#)]

다섯 글자를 배열에 넣어주세요.

그리고 0번~4번 index까지 출력 하고

4번~0번 index까지 출력하는 프로그램을

재귀호출로 만들어 주세요

## 입력 예제

```
abcde
```

## 출력 결과

```
abcde
```

```
edcba
```



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

void func(char* arr, int idx)
{
    cout << arr[idx];
    if (idx == 5)
    {
        cout << endl;
        return;
    }
    func(arr, idx + 1);
    cout << arr[idx];
}

int main(void)
{
    char input[5] = {};
    for (int i = 0; i < 5; ++i)
        cin >> input[i];

    func(input, 0);

    return 0;
}
```

# Level20 a, b 재귀호출

문제 6번 [[숙제](#) [목록보기](#)]

a, b 숫자 2개를 입력 받고,

재귀호출을 이용하여

a --> b --> a 까지 출력 해 주세요.

ex ) 3 9

3 4 5 6 7 8 9 8 7 6 5 4 3

## 입력 예제

3 9

## 출력 결과

3 4 5 6 7 8 9 8 7 6 5 4 3

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

void func(int a, int b)
{
    cout << a << " ";
    if (a == b)
        return;
    func(a + 1, b);
    cout << a << " ";
}

int main(void)
{
    int a = 0, b = 0;
    cin >> a >> b;

    func(a, b);

    return 0;
}
```

# Level20 재귀 부메랑

문제 7번 [[숙제](#) [목록보기](#)]

3	7	4	1	9	4	6	2
---	---	---	---	---	---	---	---

index 숫자 한개를 입력받으세요

해당 index부터 0번 index까지 출력 한 후

0번 index부터 입력받은 index까지 출력 하는 프로그램을 작성 해 주세요.

재귀호출을 이용하여 문제를 풀어주세요.

ex ) 3

1 4 7 3 7 4 1

## 입력 예제

3

## 출력 결과

1 4 7 3 7 4 1

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

void func(int* arr, int idx)
{
    cout << arr[idx] << " ";
    if (idx == 0)
        return;
    func(arr, idx-1);
    cout << arr[idx] << " ";
}

int main(void)
{
    int arr[8] = { 3,7,4,1,9,4,6,2 };
    int index = 0;
    cin >> index;

    func(arr,index);

    return 0;
}
```

# Level20 없어질때까지 나눠먹기

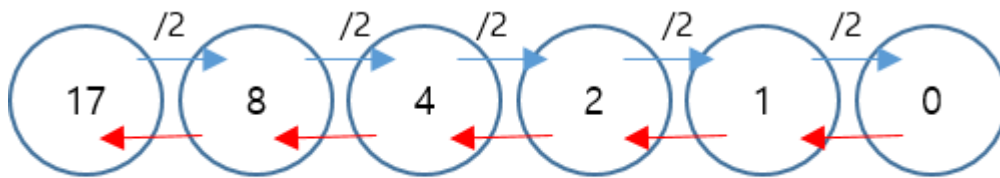
문제 8번 [숙제 [목록보기](#)]

숫자 1개를 입력받고

그 숫자가 0이 될때까지 2로 나누어 주세요

0이 된 이후에는 return하기 시작하여

되돌아 가는 값을 출력 하면 됩니다.



출력 결과 : 1 2 4 8 17

ex) 17

1 2 4 8 17

## 입력 예제

17

## 출력 결과

1 2 4 8 17

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

void func(int num)
{
    //cout << num << " ";
    if (num == 0)
        return;
    func(num / 2);
    cout << num << " ";
}

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

    func(input);

    return 0;
}
```