

(훈련반2) Level25

링크드리스트의 두 번째 시간입니다.

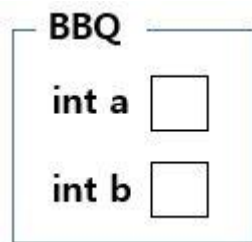
메모리 Heap 공간에 Node를 만드는 방식으로 링크드리스트를 만듭니다.

그리고 For문과 While로 탐색하여, 링크드리스트에 익숙 해 집니다.

Level25 힙에 구조체 변수 만들기

문제 1번 [[숙제 목록보기](#)]

BBQ구조체를 만들어주세요.



그리고 new연산자 또는 malloc 함수를 이용하여 힙에 변수를 만들어 주세요.

그리고 숫자 2개를 이곳에 입력 받고 5를 더한 값을 출력 해주세요.

입력 예제

1 4

출력 결과

6 9

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

struct BBQ
{
    BBQ() : a(0), b(0)
    {

    }
    int a;
    int b;
};

int main(void)
{
    BBQ* bbq = new BBQ;
    cin >> bbq->a >> bbq->b;

    int a2 = bbq->a + 5;
    int b2 = bbq->b + 5;

    cout << a2 << " " << b2;

    delete bbq;

    return 0;
}
```

Level25 new(malloc)으로 변수 만들기

문제 2번 [[숙제](#) [목록보기](#)]

new (or malloc)을 이용하여 char 변수 3개를 만들어주세요.

그리고 문자 3개를 이 변수에 입력 받고,

모두 대문자 인지 아닌지 구분하는 프로그램을 작성해주세요.

모두 대문자이면 "모두대문자"

소문자가 하나라도 있으면 "소문자있음" 이라고 출력하세요.

입력 예제

A K T

출력 결과

모두대문자

```

#include <iostream>
using namespace std;

int arr[3] = {}; // 0 : 대문자, 1 : 소문자, 2 : 알파벳 아님

void func(char* ch)
{
    if (*ch >= 65 && *ch <= 90)
        ++arr[0];
    else if (*ch >= 97 && *ch <= 122)
        ++arr[1];
    else
        ++arr[2];
}

int main(void)
{
    char* ch1 = new char;
    char* ch2 = new char;
    char* ch3 = new char;

    cin >> *ch1 >> *ch2 >> *ch3;
    // ascii
    // 'A' : 65 , 'Z' : 90
    // 'a' : 97 , 'z' : 122
    func(ch1);
    func(ch2);
    func(ch3);

    if (arr[0] == 3)
        cout << "모두대문자";
    else if (arr[0] != 3 && arr[2] == 0)
        cout << "소문자 있음";
    else
        cout << "???";

    delete ch1;
    delete ch2;
    delete ch3;

    return 0;
}

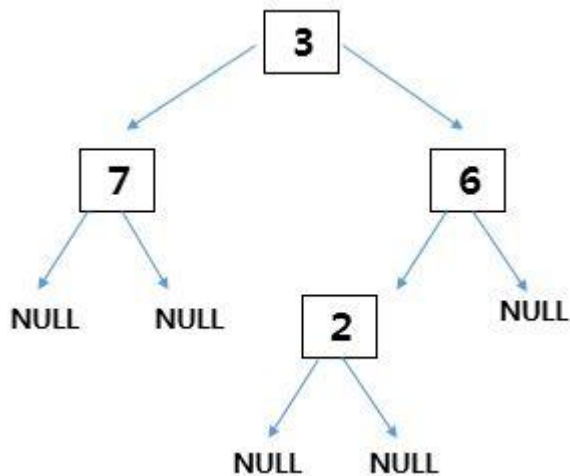
```

Level25 트리식으로 링크드리스트 구현하기

문제 3번 [숙제 [목록보기](#)]

new 또는 malloc 을 이용해서 아래의 그림처럼 연결 해주세요

(입력 및 출력 값은 없습니다)



HINT

- ① head 포인터가 필요합니다.
- ② `head -> left = new or malloc ()`
`head -> right = new or malloc ()`
`head -> right -> left = new or malloc ()`

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

struct Node
{
    Node() : val(0), left(nullptr), right(nullptr)
    {

    }
    int val;
    Node* left;
    Node* right;
};

int main(void)
{
    Node* head = nullptr;

    head = new Node;
    head->val = 3;

    head->left = new Node;
    head->left->val = 7;

    head->right = new Node;
    head->right->val = 6;

    head->right->left = new Node;
    head->right->left->val = 2;

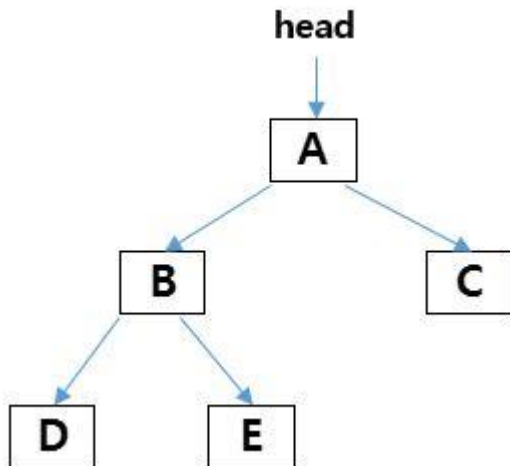
    delete head->right->left;
    delete head->right;
    delete head->left;
    delete head;

    return 0;
}
```

Level25 흔적을 쫓아가자

문제 4번 [숙제 [목록보기](#)]

다음 그림과 같이 노드를 만들어주세요.



이제 한 문장을 입력받아주세요.

H는 Head가 가르키는 값을 출력하라는 의미이고,

HR은 Head->Right 값을 출력하라는 의미입니다.

또 만약

HLL을 입력받는다면 Head->Left->Left의 값을 출력하고

HRR을 입력받는다면 Head->Right->Right 값을 출력하라는 의미입니다.

H / HR / HL / HLL / HLR에 대해 적합한 값을 출력하는 프로그램을 작성 해 주세요.

입력 예제

HLR

출력 결과

E

```

#include <iostream>

#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtdbg.h>

using namespace std;

struct Node
{
    Node() : val('\0'), right(nullptr), left(nullptr)
    {
    }
    char val;
    Node* right;
    Node* left;
};

char commandArr[5] = {};

int main(void)
{
    Node* head = nullptr;
    head = new Node;
    head->val = 'A';

    head->left = new Node;
    head->left->val = 'B';

    head->right = new Node;
    head->right->val = 'C';

    head->left->left = new Node;
    head->left->left->val = 'D';

    head->left->right = new Node;
    head->left->right->val = 'E';

    cin >> commandArr;
}

```



```
Node* p = nullptr;
for (int i = 0; i < strlen(commandArr); ++i)
{
    if (commandArr[i] == 'H')
        p = head;
    else if (commandArr[i] == 'R')
        p = p->right;
    else if (commandArr[i] == 'L')
        p = p->left;
    else
        __noop;
}

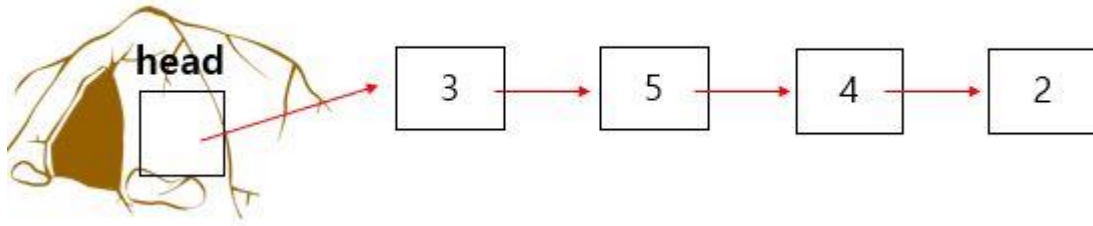
cout << p->val;

delete head->left->right;
delete head->left->left;
delete head->right;
delete head->left;
delete head;

_CrtDumpMemoryLeaks();
return 0;
}
```

Level25 동굴 탐험하기

문제 5번 [숙제 [목록보기](#)]



new / malloc을 이용해서 위와같은 노드를 구성해주세요.

이제 while문을 이용해서 처음부터 끝까지 출력 해주세요.

(입력은 없습니다)

출력 결과

3 5 4 2

```

#include <iostream>
#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtDBG.h>
using namespace std;

struct Node
{
    Node() : val(0), next(nullptr)
    {
    }
    int val;
    Node* next;
}

// 좀 더 다른 방법은 없을까...
int main(void)
{
    Node* head = nullptr;
    head = new Node;
    head->val = 3;

    head->next = new Node;
    head->next->val = 5;

    head->next->next = new Node;
    head->next->next->val = 4;

    head->next->next->next = new Node;
    head->next->next->next->val = 2;

    Node* p = head;
    while (p->next != nullptr)
    {
        cout << p->val << " ";
        p = p->next;
        if (p->next == nullptr)
            cout << p->val;
    }

    delete head->next->next->next;
    delete head->next->next;
    delete head->next;
    delete head;

    _CrtDumpMemoryLeaks();
    return 0;
}

```

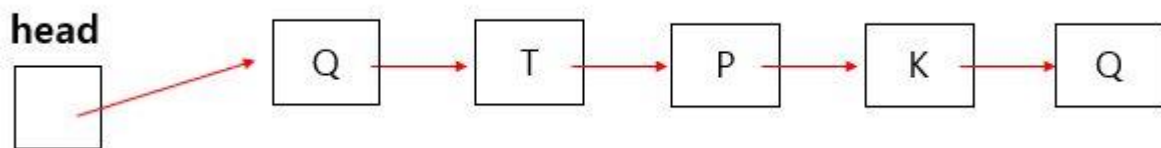
Level25 마지막 노드를 찾아라

문제 6번 [\[숙제 목록보기\]](#)

총 5개의 문자를 입력받습니다.

new 또는 malloc을 활용해, 입력받은 문자들을 링크드리스트에 저장 해 주세요.

만약 Q T P K Q 를 입력받았다면 아래와 같이 링크드리스트가 만들어집니다.



그리고 **head pointer**를 이용하여 가장 마지막 노드의 값을 출력하세요.

입력 예제

Q T P K Q

출력 결과

Q

```

#include <iostream>
#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtDBG.h>
using namespace std;

struct Node
{
    Node() : val('\0'), pNext(nullptr)
    {
    }
    char val;
    Node* pNext;
};

Node* head = nullptr;    // 첫 노드
Node* tail = nullptr;    // 마지막 노드

void addNode(char data)
{
    if (head == nullptr)
    {
        head = new Node();
        head->val = data;
        //head->pNext = nullptr;    // 이니셜라이저로 초기화
        tail = head;              // 현재 Node가 한개이다. head == tail
    }
    else
    {
        tail->pNext = new Node;
        tail->pNext->val = data;
        //tail->pNext->pNext = nullptr;    // 이니셜라이저로 초기화
        tail = tail->pNext;              // tail은 마지막 노드를 가리킨다.
    }
}

int main(void)
{
    char values[10] = {};
    cin >> values;

    for (int i = 0; i < strlen(values); ++i)
        addNode(values[i]);
    cout << tail->val << endl;
}

```

```
char lastNodeVal = '\0';
lastNodeVal = head->pNext->pNext->pNext->pNext->val;
cout << lastNodeVal;
// 메모리 해제도 함수화 할 수 없나...
delete head->pNext->pNext->pNext->pNext;
delete head->pNext->pNext->pNext;
delete head->pNext->pNext;
delete head->pNext;
delete head;

_CrtDumpMemoryLeaks();
return 0;
}
```

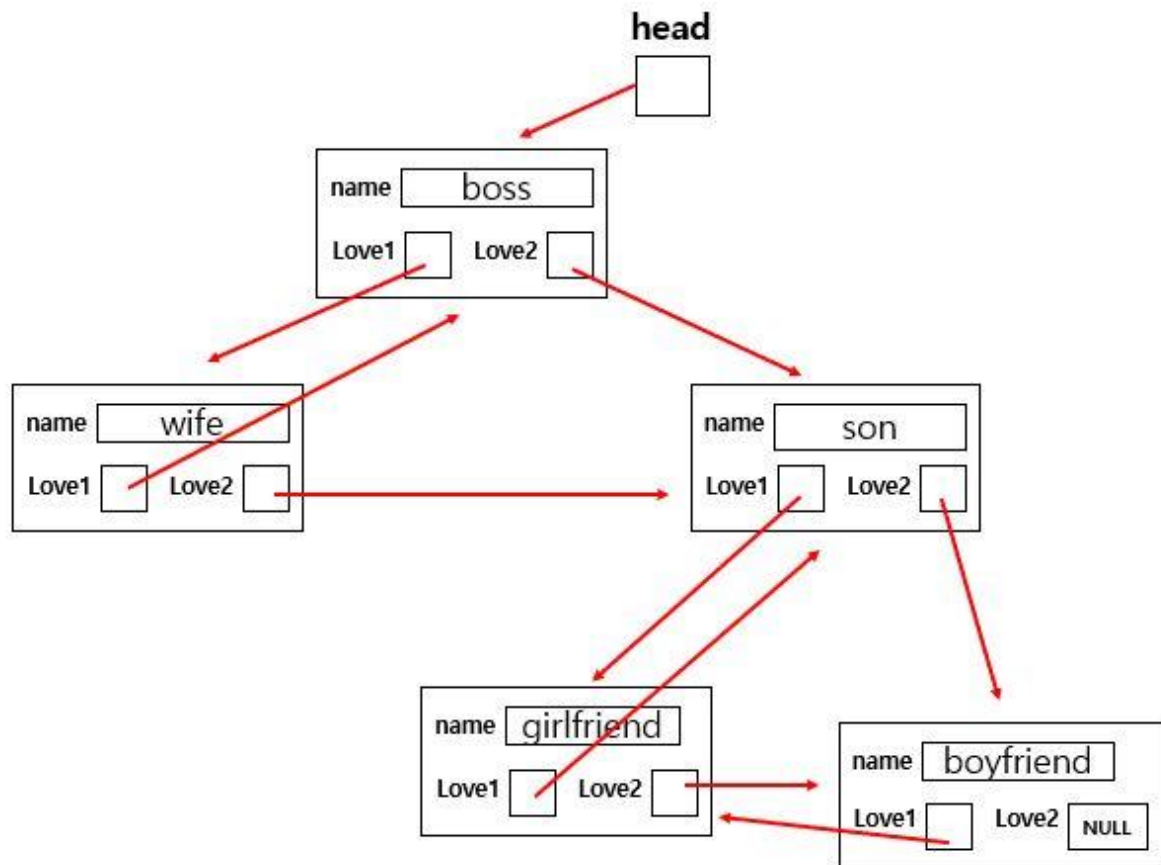
Level25 아침드라마 : 삼각관계

문제 7번 [숙제 목록보기]

아침드라마의 재미는 연인들의 관계입니다.

화살표는 사랑하는 사람을 나타낸 표식입니다.

아래와 같은 구성으로 관계가 되어 있을때 링크드리스트로 구현해주세요.



이제 head를 이용해서 son의 Love1과 Love2가 누구인지 출력 해주세요

```

#include <iostream>
#define _CRTDBG_MAP_ALLOC
#include <stdlib.h>
#include <crtDBG.h>
using namespace std;

struct Node
{
    Node() : Love1(nullptr), Love2(nullptr)
    {
    }
    Node(char* str) : Love1(nullptr), Love2(nullptr)
    {
        insertName(str);
    }
    char name[10] = {};
    Node* Love1;
    Node* Love2;

    void insertName(char* str)
    {
        for (int i = 0; i < strlen(str); ++i)
            name[i] = str[i];
    }
};

int main(void)
{
    char names[10][10] =
    {
        "boss",
        "wife",
        "son",
        "girlfriend",
        "boyfriend"
    };

    Node* head = nullptr;
    head = new Node(names[0]);           // head == boss
    head->Love1 = new Node(names[1]);    // wife
    head->Love2 = new Node(names[2]);    // son

    head->Love1->Love1 = head;           // wife -(Love1)-> boss
    head->Love1->Love2 = head->Love2;    // wife -(Love2)-> son

    head->Love2->Love1 = new Node(names[3]); // girlfriend
    head->Love2->Love2 = new Node(names[4]); // boyfriend
}

```



```
head->Love2->Love1->Love1 = head->Love2;           // girlfriend -(Love1)-> son
head->Love2->Love1->Love2 = head->Love2->Love2;      // girlfriend -(Love2)-> boyfriend

cout << "son's Love1 : " << head->Love2->Love1->name << endl;
cout << "son's Love2 : " << head->Love2->Love2->name << endl;

delete head->Love2->Love2;
delete head->Love2->Love1;
delete head->Love2;
delete head->Love1;
delete head;

_CrtDumpMemoryLeaks();
return 0;
}
```

Level25 조합한 숫자에 5 더하기

문제 8번 [숙제 목록보기]

숫자 하나만 입력 받으세요. (int 변수에다가 숫자 하나 입력받으세요)

6자리 숫자만 입력할 수 있습니다.

두 번째 자리 숫자와 네 번째 자리 숫자를 추출하고 조합하세요

그리고 만들어진 두 자리 숫자에 5를 더한 값을 출력 해주세요.

ex)

입력 : 354129

출력 : 47



입력 예제

354129

출력 결과

47

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

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

    int second = num % 100 - num % 10;
    int fourth = (num % 10000 - num % 1000);
    int combination = fourth / 100 + second / 10;
    int result = combination + 5;
    cout << result;

    return 0;
}
```

Level25 내려오는 물방울 갯수 세기

문제 9번 [\[숙제 목록보기\]](#)

4 x 4 배열을 입력받아주세요. #은 빈칸을 의미합니다.

각 세로줄마다 몇개의 문자가 있는지 출력 하세요.

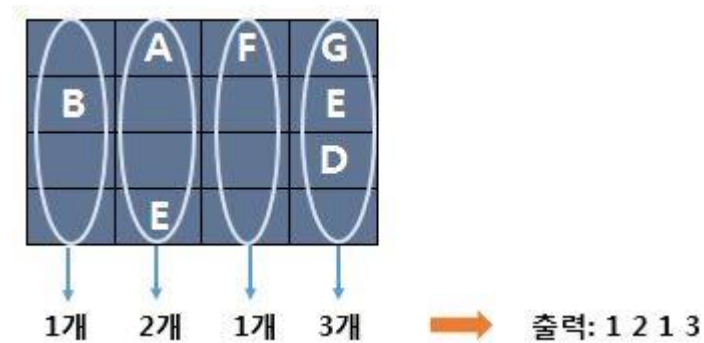
만약 아래와 같이 입력받았다면,

#AFG

B##E

###D

#E##



입력 예제

#AFG

B##E

###D

#E##

출력 결과

1 2 1 3

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

const int sizeY = 4;
const int sizeX = 4;

int main(void)
{
    char str2D[sizeY][sizeX] = {};
    for (int y = 0; y < sizeY; ++y)
    {
        for (int x = 0; x < sizeX; ++x)
        {
            cin >> str2D[y][x];
        }
    }

    for (int x = 0; x < sizeX; ++x)
    {
        int cnt = 0;
        for (int y = 0; y < sizeY; ++y)
        {
            if (str2D[y][x] != '#' && str2D[y][x] != '\\0')
                ++cnt;
        }
        cout << cnt << " ";
    }

    return 0;
}
```

Level25 짝궁

문제 10번 [[숙제](#) [목록보기](#)]

B와 F로 된 문장을 입력 받고 짝이 맞는지 알려주는 프로그램을 작성해주세요.

(최대 글자수는 10글자)

B는 Begin을 뜻하고(시작) F는 Finish를 뜻합니다.(종료)

그리고 B와 F는 서로 짝이 맞아야 합니다.

아래 내용을 참고하세요.

ex)

BFBFBF	→	짝이 맞음
FBBF	→	짝이 안 맞음(B보다 F가 먼저 시작함)
BBBFBFFF	→	짝이 맞음
BBFBF	→	짝이 안 맞음(F가 하나 부족)

BBFF인 경우는 짝이 맞습니다

시작이 2번 되었으니 끝 신호가

2번 와야 하기 때문입니다

짝이 맞으면 "짝이맞음" 출력

안맞으면 "짝이안맞음" 출력

입력 예제

BFBFBF

출력 결과

짝이맞음

```

#include <iostream>
using namespace std;

// 짝이 맞다?
// 1. B와 F의 개수가 같아야 한다.
// 2. F보다 B가 먼저 나와야 한다.
// 3. B의 총 개수와 F의 총 개수가 같을 때
//     B가 F보다 먼저 많이 나와도 된다.
//     ex) BBBFFF, BBFBFF

const int inputCnt = 10;

int main(void)
{
    char input[inputCnt + 1] = {};    // 최대 10글자
    cin >> input;

    int bCnt = 0;
    int fCnt = 0;
    bool isPair = false;

    for (int i = 0; i < inputCnt; ++i)
    {
        if (input[i] == 'B')
            ++bCnt;
        if (input[i] == 'F')
            ++fCnt;
        if (bCnt < fCnt)
            break;
        if (input[i] == '\\0')
        {
            if (bCnt == fCnt)
                isPair = true;
            break;
        }
    }

    if (isPair)
        cout << "짝이맞음";
    else
        cout << "짝이안맞음";

    return 0;
}

```

Level25 블럭 굴리기

문제 11번 [숙제 목록보기]

아래 그림과 같이 3 x 3의 네모블럭이 있습니다. 하드코딩 하세요.

	5	4
3		
		1

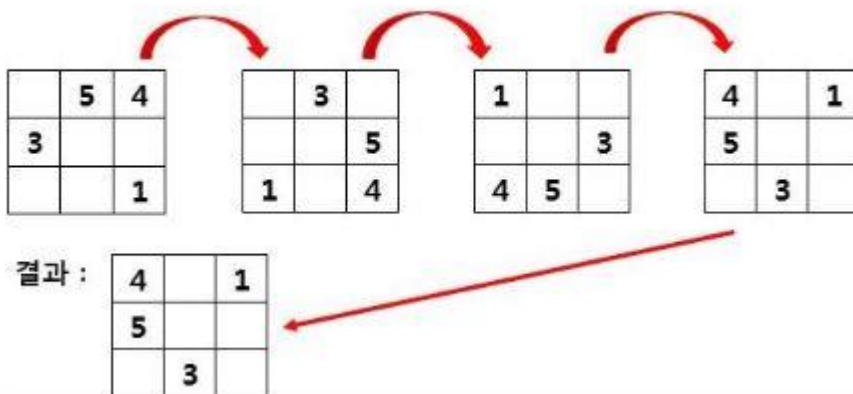
그리고 숫자 하나를 입력 받으세요.

이 입력 숫자는 블럭을 오른쪽으로 굴릴 횟수 입니다.

입력한 숫자만큼 오른쪽으로 굴려 숫자 위치가 변경된 상태를 출력 해주세요.

*** 빈칸은 언더바('_')로 바꾸어서 출력 해 주세요**

Ex) 굴릴 횟수 : 3



[HINT]

오른쪽으로 한번 굴리면 좌표가 어떻게 바뀌는지 써 보세요

숫자가 들어있는 cube[3][3] 배열이 있고

돌린 결과를 저장할 result[3][3] 배열을 한개 더 만든다.

이동할 좌표 (y, x) (result[3][3] <-- cube[3][3])

```
0,2 <-- 0,0
```

```
1,2 <-- 0,1
```

```
2,2 <-- 0,2
```

```
0,1 <-- 1,0
```

```
1,1 <-- 1,1
```

```
2,1 <-- 1,2
```

```
0,0 <-- 2,0
```

```
1,0 <-- 2,1
```

```
2,0 <-- 2,2
```

위 이동 좌대로 for문을 돌리면

```
for (y=0; y<3; y++) {  
    for (x=0; x<3; x++) {  
        result[_][_] = input[y][x];  
    }  
}
```

입력 예제

```
3
```

출력 결과

```
4_1
```

```
5__
```

```
_3_
```



```

#include <iostream>
using namespace std;

const int blockSize = 3;
const int blockCnt = 9;
int block[blockSize][blockSize] =
{
    0,5,4,
    3,0,0,
    0,0,1
};

void rotate(int rotateCnt)
{
    int tempBlock[blockSize][blockSize] = {};
    int cnt = 0;
    while (cnt < rotateCnt)
    {
        int idx = 0;
        for (int x = blockSize - 1; x >= 0; --x)
        {
            for (int y = 0; y < blockSize; ++y)
            {
                int val = block[idx / blockSize][idx % blockSize];
                tempBlock[y][x] = val;
                ++idx;
            }
        }

        for (int i = 0; i < blockCnt; ++i)
            block[i / blockSize][i % blockSize] = tempBlock[i / blockSize][i %
blockSize];

        ++cnt;
    }
}

```

```
void printBlock()
{
    for (int y = 0; y < blockSize; ++y)
    {
        for (int x = 0; x < blockSize; ++x)
        {
            if (block[y][x] == 0)
                cout << '_';
            else
                cout << block[y][x];
        }
        cout << endl;
    }
}

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

    rotate(rRotateCnt);    // 회전 로직
    printBlock();          // 출력

    return 0;
}
```

Level25 배열 만들어 값 넣기

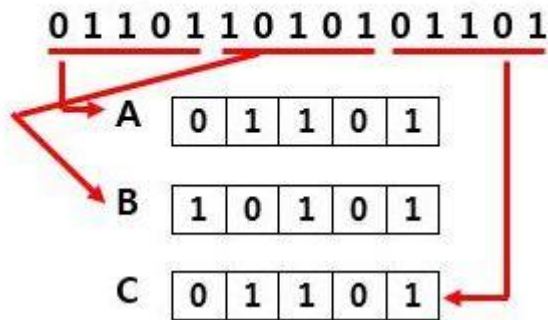
문제 12번 [[숙제](#) [목록보기](#)]

2진수형태의 한 문장을 입력 받으세요.(15글자)

그리고 5자리씩 끊어서

배열 3개에 넣고 출력 해주세요.

Ex)



입력: 011011010101101 (문자열로 입력)

출력: 01101

10101

01101

입력 예제

011011010101101

출력 결과

01101

10101

01101

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

const int strMaxLength = 15;
const int printArrY = 3;
const int printArrX = 5;

int main(void)
{
    char str[strMaxLength + 1] = {}; // 최대 15글자
    cin >> str;

    char printArr[printArrY][printArrX] = {};

    for (int i = 0; i < strlen(str); ++i)
    {
        printArr[i / printArrX][i % printArrX] = str[i];
    }

    for (int k = 0; k < strlen(str); ++k)
    {
        cout << printArr[k / printArrX][k % printArrX];
        if (k % 5 == 4)
            cout << endl;
    }
    return 0;
}
```

Level25 min과 max 찾기

문제 13번 [[숙제](#) [목록보기](#)]

Number	3	7	4	0	9	6
--------	---	---	---	---	---	---

Command	m	x	m	m	x	x
---------	---	---	---	---	---	---

6개의 숫자와 m과 x로 된 한 문장을 입력 받아주세요.

m은 min값을 의미하고, x는 max값을 의미합니다.

Command[0]부터~[5]까지 순서대로 탐색하면서

m이면 min값을, x면 max값을 Number배열에서 찾아 출력 하세요.

* Number에 같은 숫자는 입력되지 않습니다.

* 한번 출력한 숫자는 다시 출력하지 않습니다.

입력 예제

3 7 4 0 9 6

mxmxx

출력 결과

093476

```

#include <iostream>
using namespace std;

const int NumberSize = 6;
const int CommandSize = 6;

int Number[NumberSize] = {};
char Command[CommandSize + 1] = {};    // 67H
bool isVisit[NumberSize] = {};

void ascSort()
{
    for (int i = 0; i < NumberSize - 1; ++i)
    {
        for (int k = i + 1; k < NumberSize; ++k)
        {
            if (Number[i] > Number[k])
            {
                int temp = Number[i];
                Number[i] = Number[k];
                Number[k] = temp;
            }
        }
    }
}

int main(void)
{
    for (int i = 0; i < NumberSize; ++i)
        cin >> Number[i];
    cin >> Command;

    ascSort();
    int minIdx = 0;
    int maxIdx = NumberSize - 1;
    for (int k = 0; k < CommandSize; ++k)
    {
        if (Command[k] == 'm')
        {
            cout << Number[minIdx];
            ++minIdx;
        }
        else if (Command[k] == 'x')
        {
            cout << Number[maxIdx];
            --maxIdx;
        }
    }
}

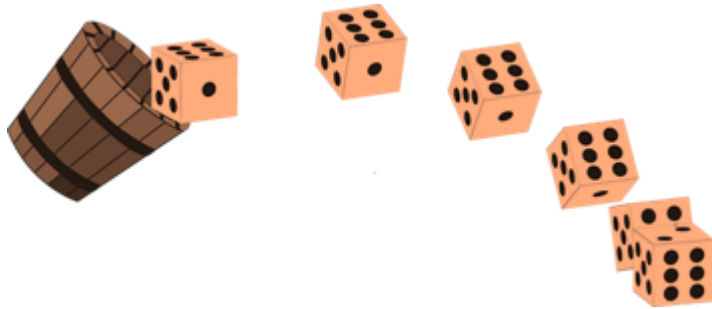
```

```
        else
            __noop;
    }

    return 0;
}
```

Level25 주사위 던지기

문제 14번 [[숙제](#) [목록보기](#)]



주사위 n 개로 나올 수 있는 눈금을 모두 출력 해주세요.

ex) 주사위 개수 3 입력 받으면

111

112

113

114

115

116

121

122

123

124

...

666

입력 예제

2

출력 결과

11

12

13

14

15

16

21

22

23

24

25

26

31

32

33

34

35

36

41

42

43

44

45

46

51

52

53

54

55

56

61

62

63

64

65

66

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

int DiceCnt = 0;           // 주사위 개수
const int markingCnt = 6; // 주사위 눈금 수
char markCombi[10] = {};  // 주사위 조합
const int initLevel = 0;  // 시작 레벨
const int offset = 49;    // ascii -> '1' = 49 , '6' = 54
int caseCnt = 0;          // 주사위 조합 수

void recursive(int level)
{
    if (level == DiceCnt)
    {
        ++caseCnt;
        cout << markCombi << endl;
        return;
    }
    for (int i = 0; i < markingCnt; ++i)
    {
        markCombi[level] = i + offset;
        recursive(level + 1);
        markCombi[level] = 0;
    }
}

int main(void)
{
    cin >> DiceCnt;
    recursive(initLevel);
    cout << "주사위 조합 수 : " << caseCnt;

    return 0;
}
```