

브루트 포스 - 문제

최백준 choi@startlink.io

차량 번호판 1

<https://www.acmicpc.net/problem/16968>

- 차량 번호판은 길이가 4이하이면서, c와 d로 이루어진 문자열이다.

c는 문자, d는 숫자

26개

- 사용할 수 있는 문자는 a, b, c, ..., y, z

알파벳 26개

- 사용할 수 있는 숫자는 0, 1, 2, ..., 8, 9

10개

- 같은 문자나 숫자가 두 번 연속해서 나타나면 안된다.

- 가능한 차량 번호판의 개수를 구하는 문제

3개

CCCC

26⁴

↓↓↓↓
↑↑↑↑
15⁴

C D
↑ ↑

차량 번호판 1

<https://www.acmicpc.net/problem/16968>

- 가능한 경우의 수는 $26^4 = 456,976$ 이하이다.
- 전체 경우의 수를 살펴보는 것이 가능하다.

$$O(26^4)$$

차량 번호판 1

<https://www.acmicpc.net/problem/16968>

4

```
int go(string &s, int index, char last) {  
    if (s.length() == index) {  
        return 1;  
    }  
    char start = (s[index] == 'c' ? 'a' : '0');  
    char end = (s[index] == 'c' ? 'z' : '9');  
    int ans = 0;  
    for (char i=start; i<=end; i++) {  
        if (i != last) {  
            ans += go(s, index+1, i);  
        }  
    }  
    return ans;  
}
```

내보내기

Index 41212
288292 222

412 01 (012)

Index-1 4121

1/2/3/4 2/2/2/2

C: a-z
: 0-9

이제: go(s, 0, '')

62714

5

차량 번호판 1

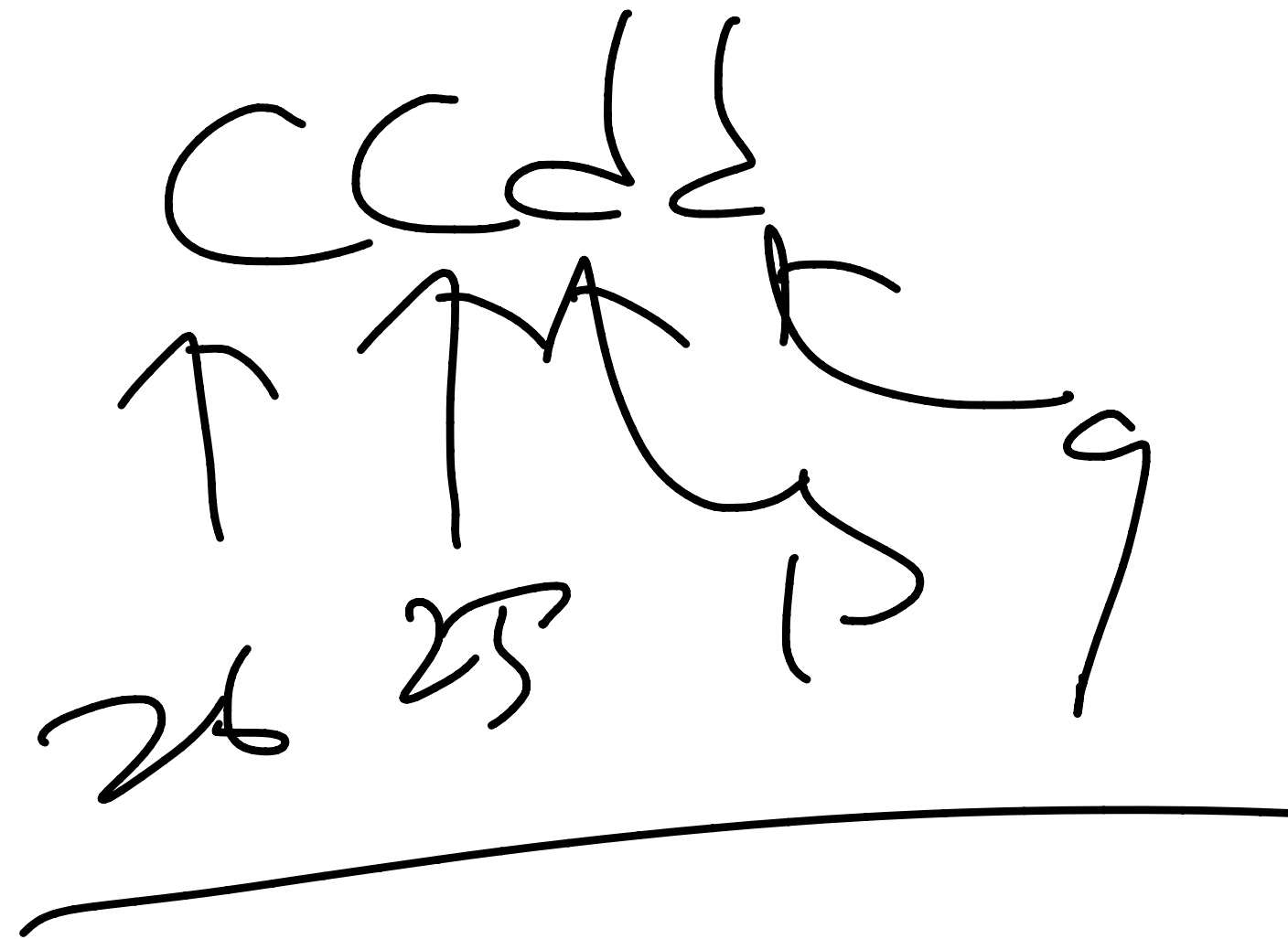
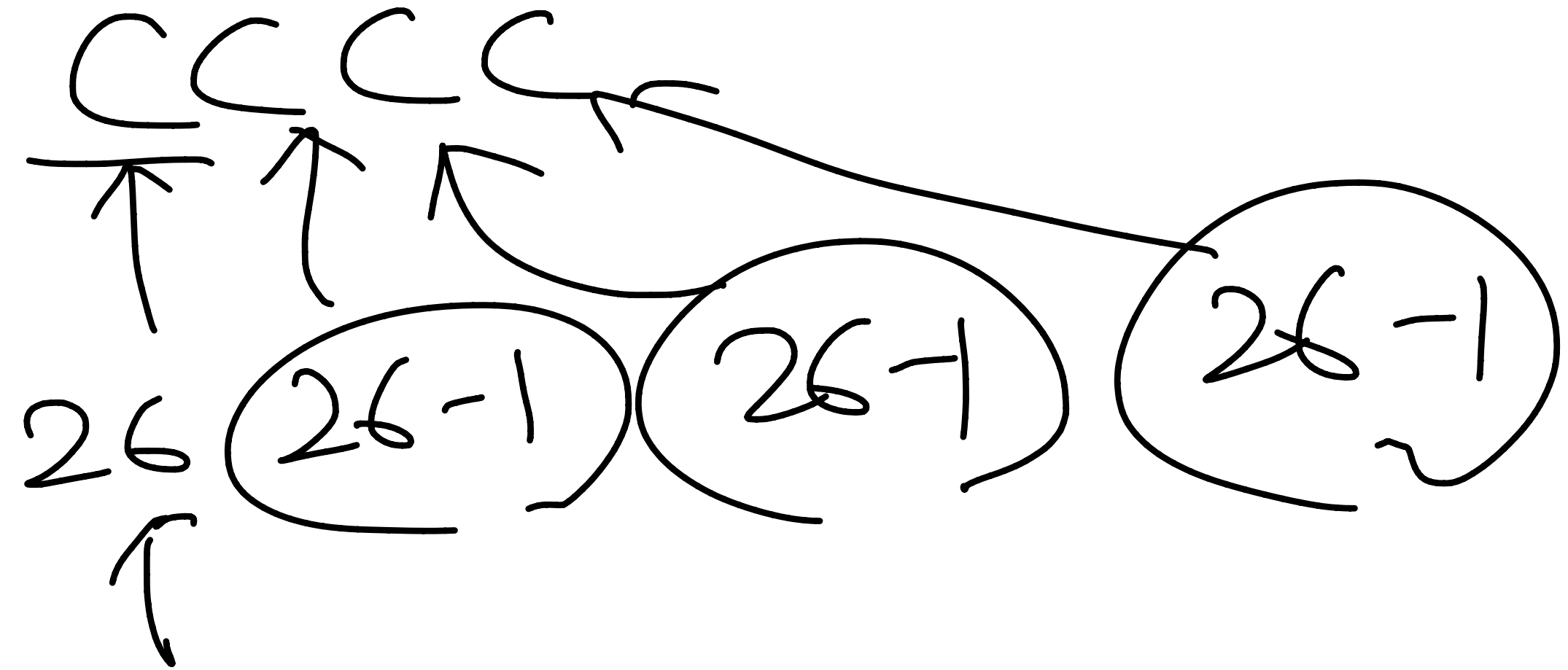
<https://www.acmicpc.net/problem/16968>

- 소스: <http://codeplus.codes/aac7d0c26a6446cab59c3e0ecaafc934d>

차량 번호판 1

<https://www.acmicpc.net/problem/16968>

- 조합을 이용해서도 해결할 수 있다.

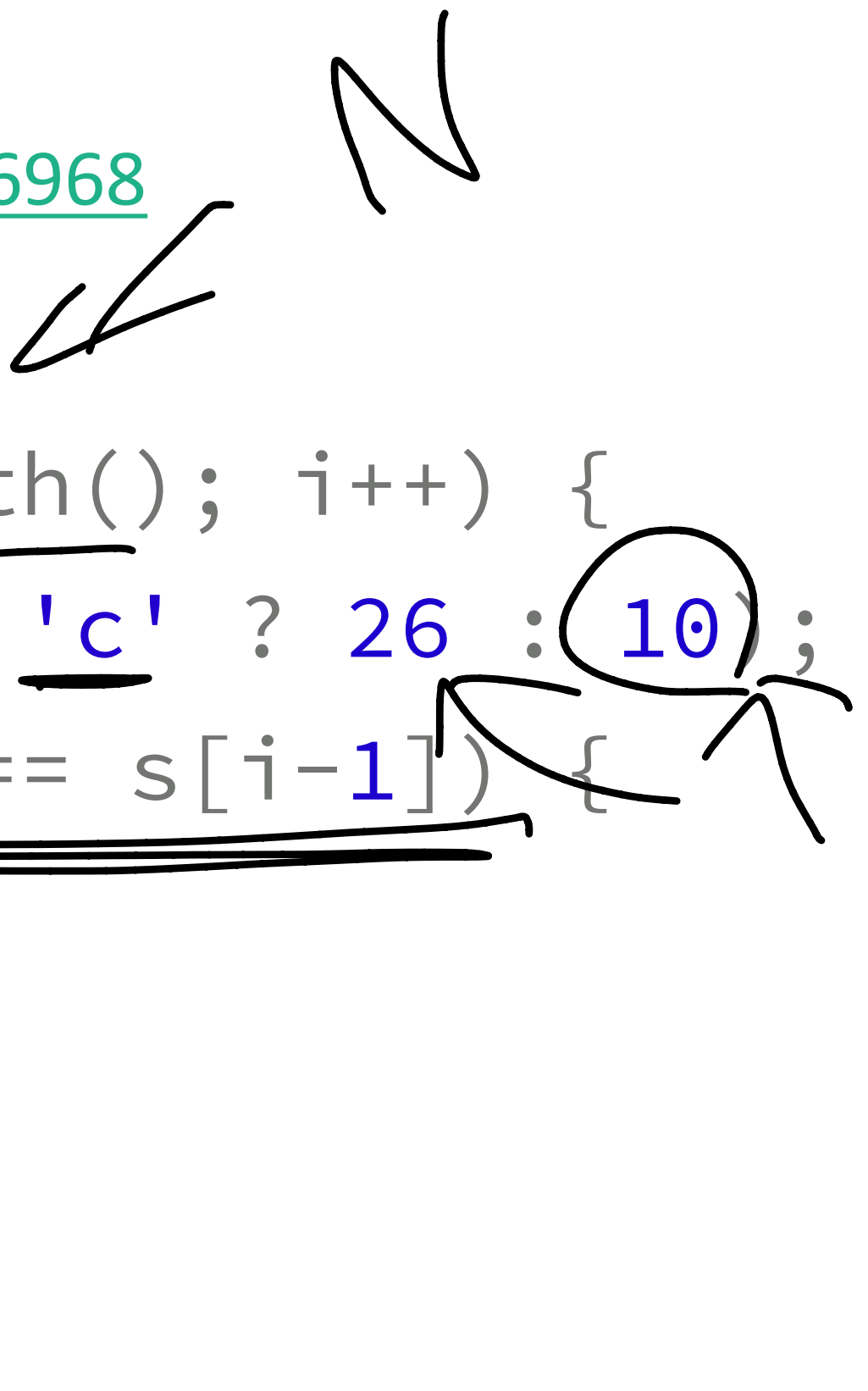


차량 번호판 1

7

<https://www.acmicpc.net/problem/16968>

```
int ans = 1;
for (int i=0; i<s.length(); i++) {
    int cnt = (s[i] == 'c' ? 26 : 10);
    if (i > 0 && s[i] == s[i-1]) {
        cnt -= 1;
    }
    ans = ans * cnt;
}
```



$O(N)$

차량 번호판 1

<https://www.acmicpc.net/problem/16968>

- 소스: <http://codeplus.codes/b4bbaeee589041c4802b7c60111d4200>

양념 반 후라이드 반

<https://www.acmicpc.net/problem/16917>

• 양념 치킨의 가격: A원

• 후라이드 치킨의 가격: B원

• 반반 치킨(양념 반 후라이드 반)의 가격: C원

• 반반 치킨을 2마리 구매해서, 양념 치킨 1마리, 후라이드 치킨 1마리를 만들 수 있다.

• 양념 치킨 최소 X마리, 후라이드 치킨 최소 Y마리를 구매하는 가장 싼 비용을 구하는 문제

• $1 \leq A, B, C \leq 5,000$

• $1 \leq X, Y \leq 100,000$

정답:

$$AX + B$$

정답

결과

2개

반반 치킨

22개

3/4 20만개

양념 반 후라이드 반

10

<https://www.acmicpc.net/problem/16917>

- $A = 1500, B = 2000, C = 1600, X = 3, Y = 2$
- 반반 치킨 4마리를 구매해서, 양념 치킨 2마리와 후라이드 치킨 2마리를 만들고, 양념 치킨 1마리를 구매하는 것이 최소
- 답: 7900

양념 반 후라이드 반

<https://www.acmicpc.net/problem/16917>

- $1 \leq X, Y \leq 100,000$
- 양념 치킨을 최대 100,000개, 후라이드 치킨을 최대 100,000개, 반반 치킨을 최대 200,000개 구매할 수 있다.
- 구매할 반반 치킨의 수를 정해놓으면, 나머지 치킨을 몇 개 구매해야하는지 구할 수 있다.

양념 반 후라이드 반

<https://www.acmicpc.net/problem/16917>

- 반반 치킨을 $2i$ 개 구매했다면
- 양념 치킨은 $X-i$ 개, 후라이드 치킨은 $Y-i$ 개 구매해야 한다.
- 이때 비용은

$$2i \times C + \max(0, X-i) \times A + \max(0, Y-i) \times B$$

- 가능한 i 의 범위는 $0 \leq i \leq 100,000$

양념

반

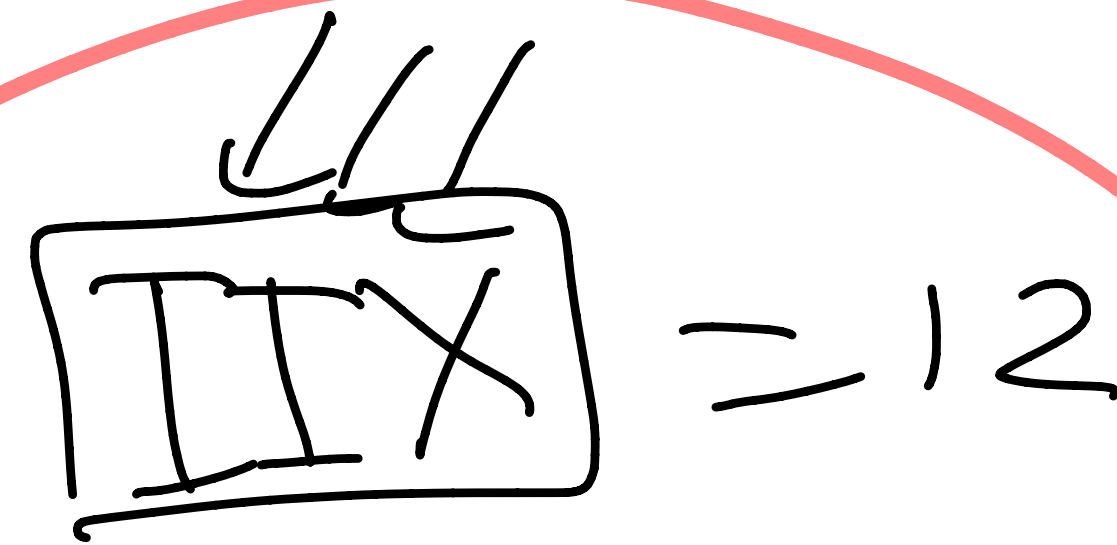
후

양념 반 후라이드 반

<https://www.acmicpc.net/problem/16917>

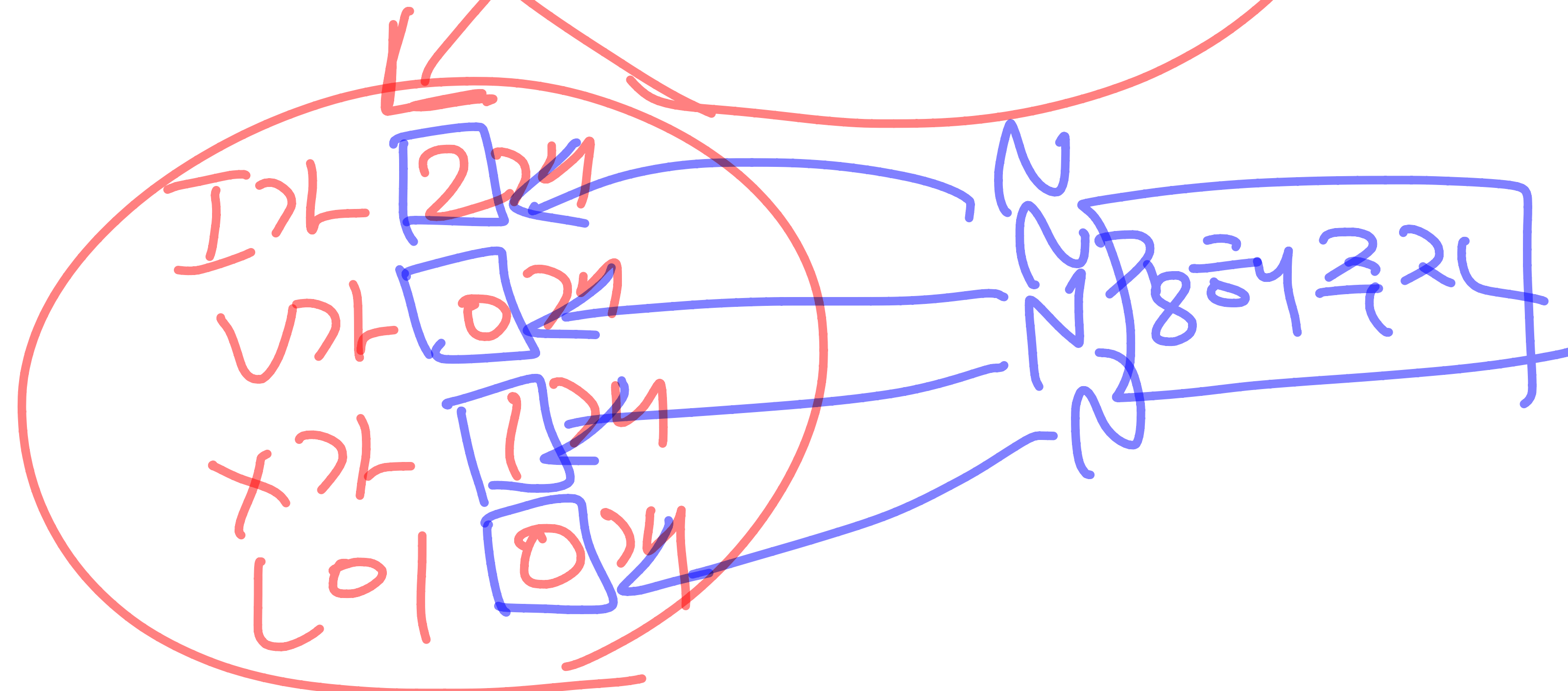
- 소스: <http://codeplus.codes/96ef289a35274990b69258862b5decea>

13


$$\boxed{\text{IIX}} = 12$$

$$\text{IXI} = 12$$

$$\text{XII} = 12$$

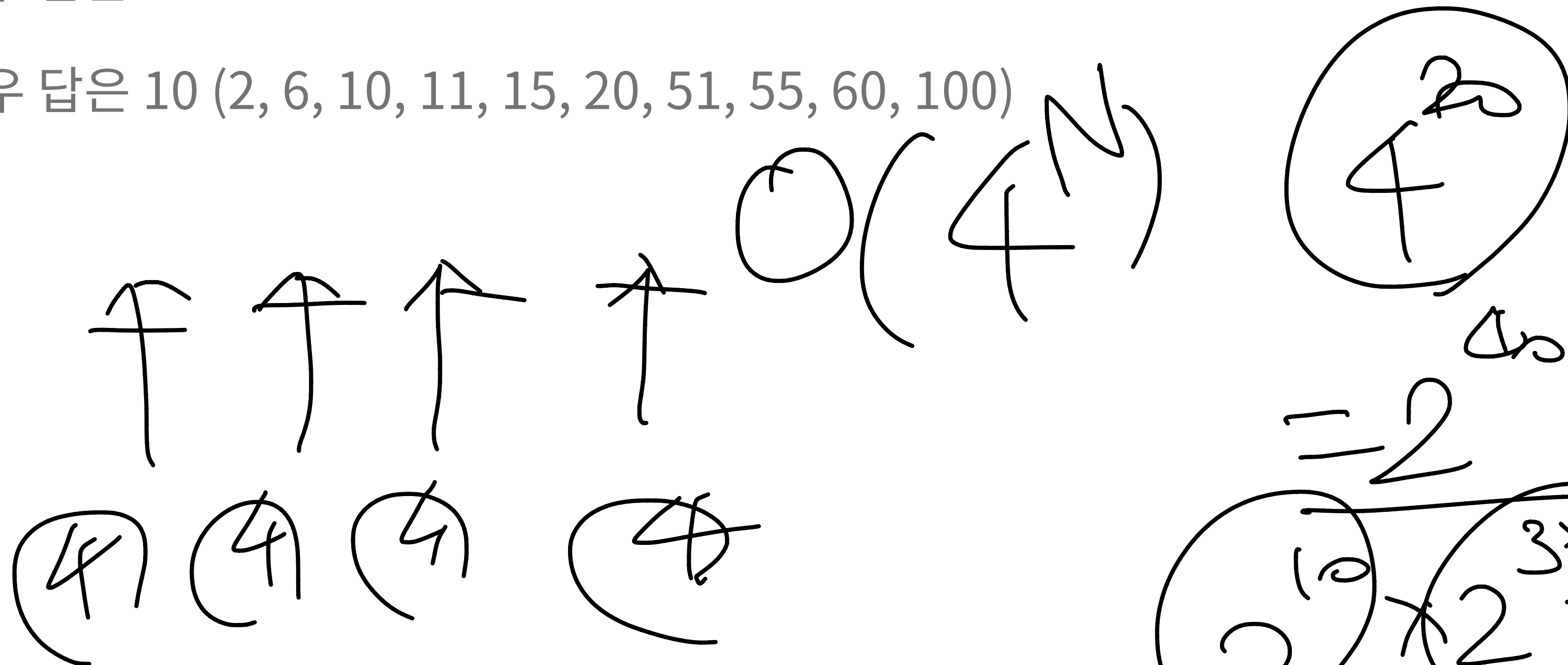
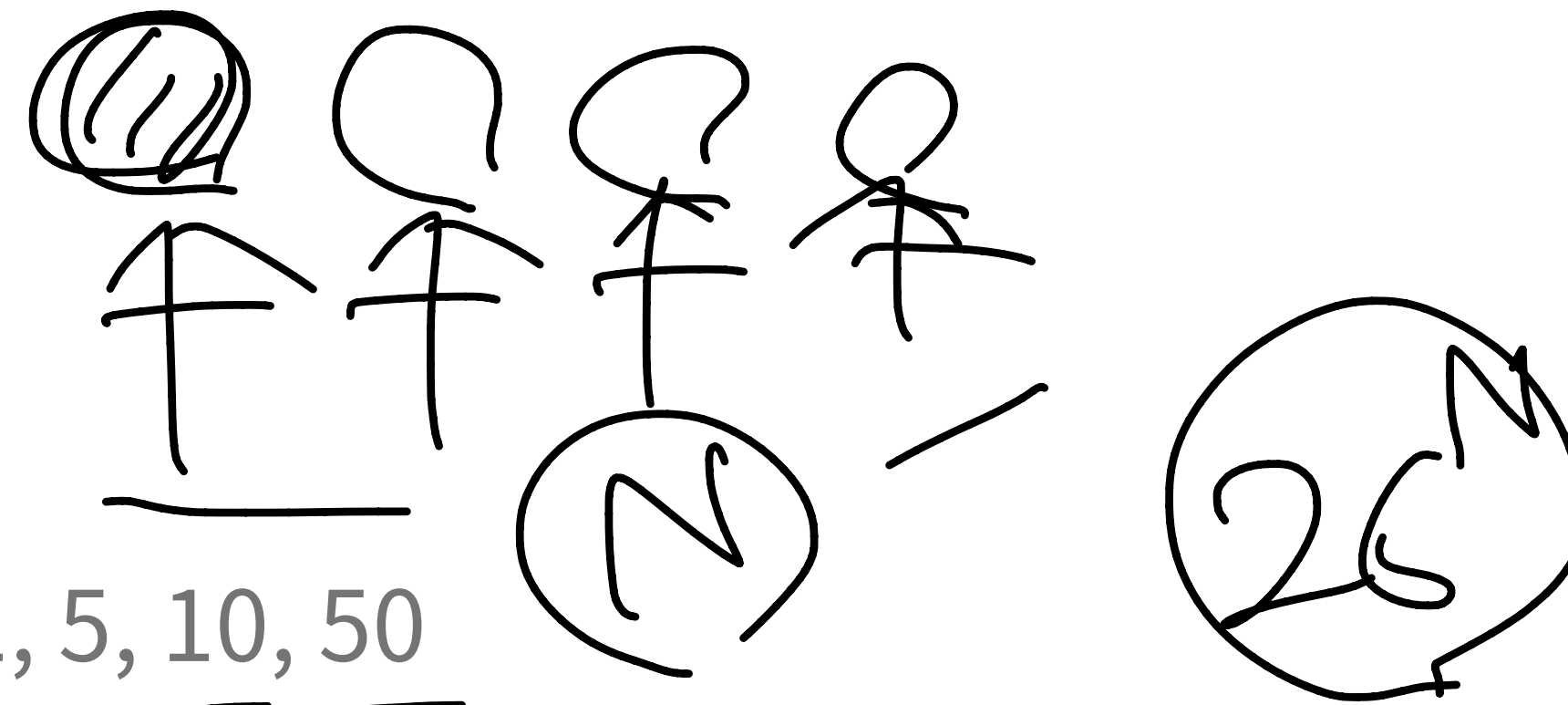


로마 숫자 만들기

<https://www.acmicpc.net/problem/16922>

14

- 로마 숫자는 I, V, X, L을 사용한다. 각각의 값은 1, 5, 10, 50
- 로마 숫자를 N 개 사용해서 만들 수 있는 서로 다른 수의 개수를 구하는 문제 ($N \leq 20$)
- $N = 1$ 인 경우 답은 4
- $N = 2$ 인 경우 답은 10 (2, 6, 10, 11, 15, 20, 51, 55, 60, 100)



$$2^{10} \times 2^{30} = 1000 \times 1000 = 10^6$$

로마 숫자 만들기

<https://www.acmicpc.net/problem/16922>

- 로마 숫자는 I, V, X, L을 사용한다. 각각의 값은 1, 5, 10, 50
- 로마 숫자를 N개 사용해서 만들 수 있는 서로 다른 수의 개수를 구하는 문제 ($N \leq 20$)
- 경우의 수: 4^N 가지

로마 숫자 만들기

<https://www.acmicpc.net/problem/16922>

- 로마 숫자는 I, V, X, L을 사용한다. 각각의 값은 1, 5, 10, 50
- 로마 숫자를 N개 사용해서 만들 수 있는 서로 다른 수의 개수를 구하는 문제 ($N \leq 20$)
- 경우의 수: 4^N 가지
- 가 아니다.
- 순서만 다른 것은 의미가 없기 때문에, 경우의 수는 N^4 가지이다.

$20^4 = 40000$

I V X L

$$\frac{N!}{a!b!c!} = N - a - b - c$$

로마 숫자 만들기

<https://www.acmicpc.net/problem/16922>

- 로마 숫자는 I, V, X, L을 사용한다. 각각의 값은 1, 5, 10, 50.
- 로마 숫자를 N개 사용해서 만들 수 있는 서로 다른 수의 개수를 구하는 문제 ($N \leq 20$)
- 경우의 수: 4^N 가지
- 가 아니다.
- 순서만 다른 것은 의미가 없기 때문에, 경우의 수는 N^4 가지이다.
- I, V, X의 개수를 알고 있다면, L의 개수도 알 수 있기 때문에, 경우의 수는 N^3 가지이다.

$$\frac{L \dots L}{20^4} = 2 \times 50 = 100$$

로마 숫자 만들기

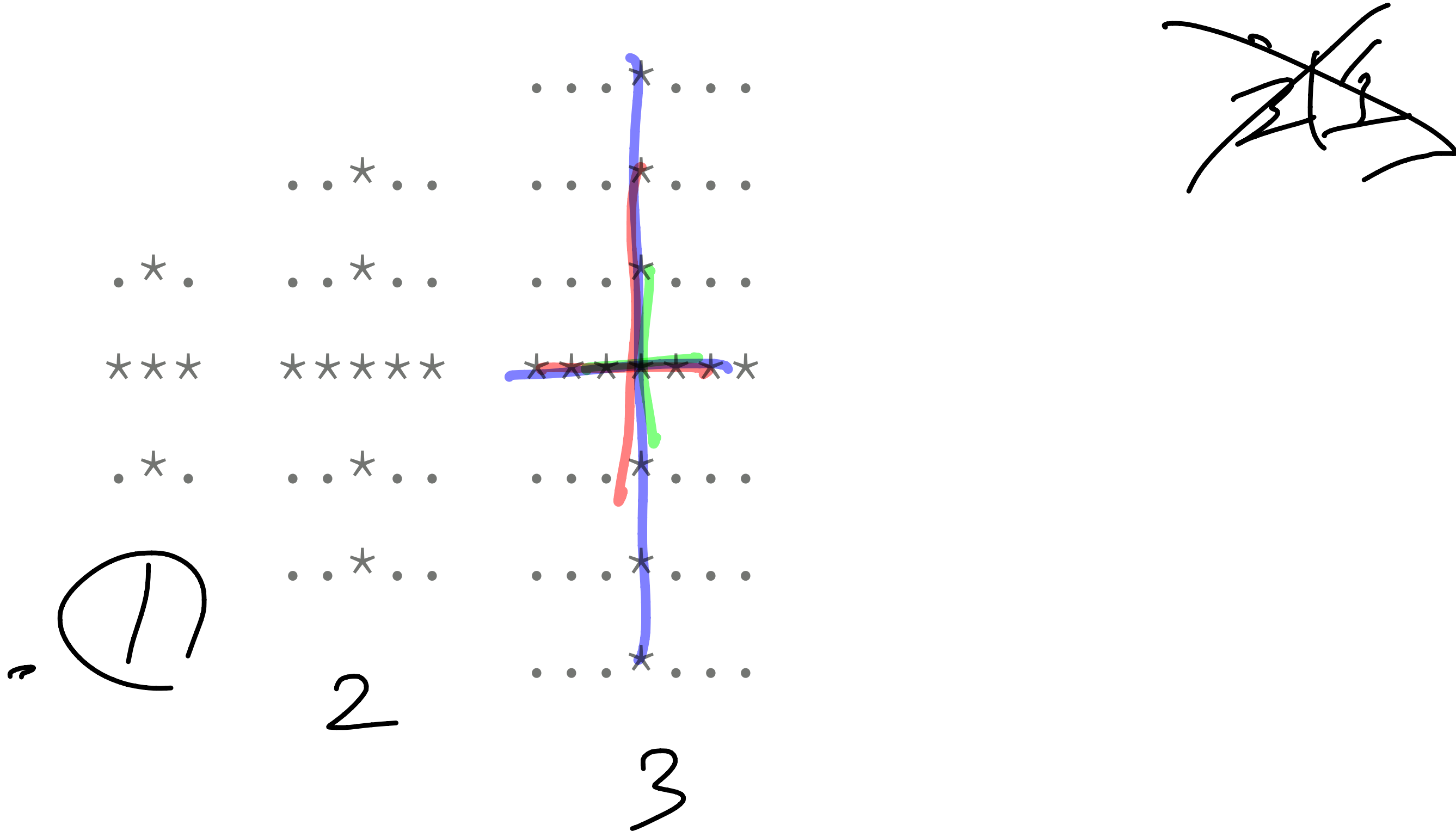
<https://www.acmicpc.net/problem/16922>

- 소스: <http://codeplus.codes/7b52510fb0c24b79becc6ffec866fe0c>

십자가 찾기

<https://www.acmicpc.net/problem/16924>

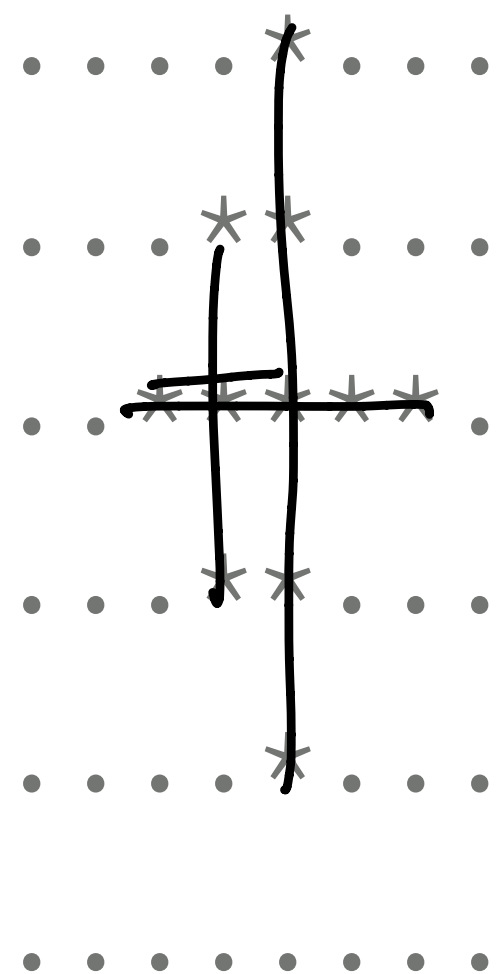
- 십자가는 가운데 *이 있고, 상하좌우로 s개의 *이 연속되는 모양이다. 이때, s가 십자가의 크기
- 격자판이 주어졌을 때, 주어진 격자판을 십자가만 이용해서 만들 수 있는 방법을 아무거나 찾는 문제



십자가 찾기

20

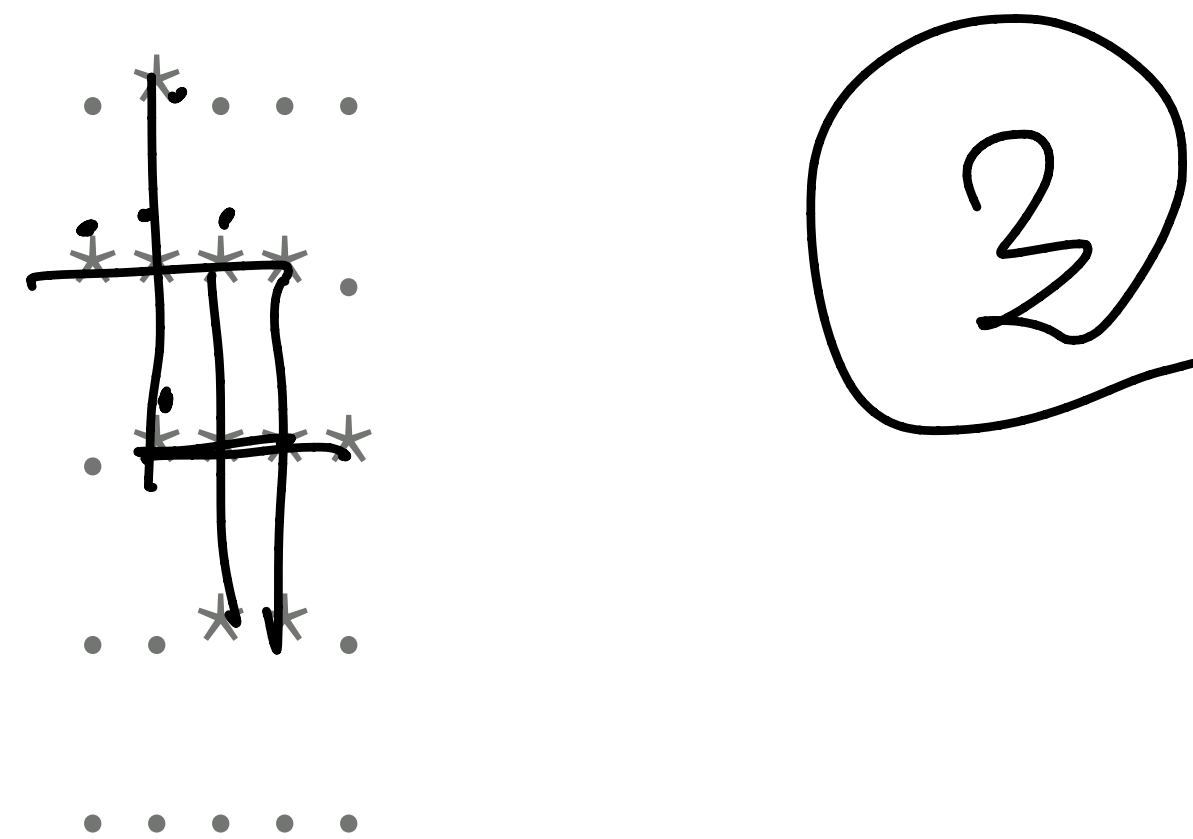
<https://www.acmicpc.net/problem/16924>



2개

십자가 찾기

<https://www.acmicpc.net/problem/16924>



십자가 찾기

$$N, M \leq 100$$

22

<https://www.acmicpc.net/problem/16924>

- 사용할 수 있는 십자가 개수는 NM 개보다 작거나 같기 때문에
- 모든 $*$ 를 십자가의 중앙이라고 가정하고, 최대한 크게 그려보면 된다.

십자가 찾기

<https://www.acmicpc.net/problem/16924>

- 소스: <http://codeplus.codes/0a11312edef941eda1944843e2fd7062>

나3곱2

<https://www.acmicpc.net/problem/16936>

24

• 정수 x 로 시작해서, 연산을 $N-1$ 번 사용해 수열 A 를 만든다

• 나3: x 를 3으로 나눈다. x 는 3으로 나누어 떨어져야 한다.

• 곱2: x 에 2를 곱한다.

• 수열 A 의 순서를 섞은 수열 B 가 주어졌을 때, A 를 구하는 문제

• $B = [4, 8, 6, 3, 12, 9]$ 인 경우 $A = [9, 3, 6, 12, 4, 8]$

A_1, A_{i+1}, \dots

9 3 6 12 4 8
2x 1x 1x 0x

$N \leq 100$

$12 = 2^2 \times 3$
 $9 = 3^2$

4 = 2^2
8 = 2^3
6 = 2×3
3 = 3

연산
3
2

나3곱2

25

<https://www.acmicpc.net/problem/16936>

$O(N \lg N)$

- 항상 가능한 입력만 주어진다.
- 각각의 수가 3으로 몇 번 나누어 질 수 있는지 기록을 한다. i 번째 수가 3으로 나누어지는 횟수를 $D[i]$ 라고 하자.
- 정답 수열 A 에서 $D[A[i]] \geq D[A[i+1]]$ 을 만족해야 한다.
- 같은 값을 갖는다면, $A[i] * 2 = A[i+1]$ 을 만족해야 한다.

나3곰2

<https://www.acmicpc.net/problem/16936>

- 소스: <http://codeplus.codes/576d2c4c59e04a72aaad182574b0f1de>

두 스티커

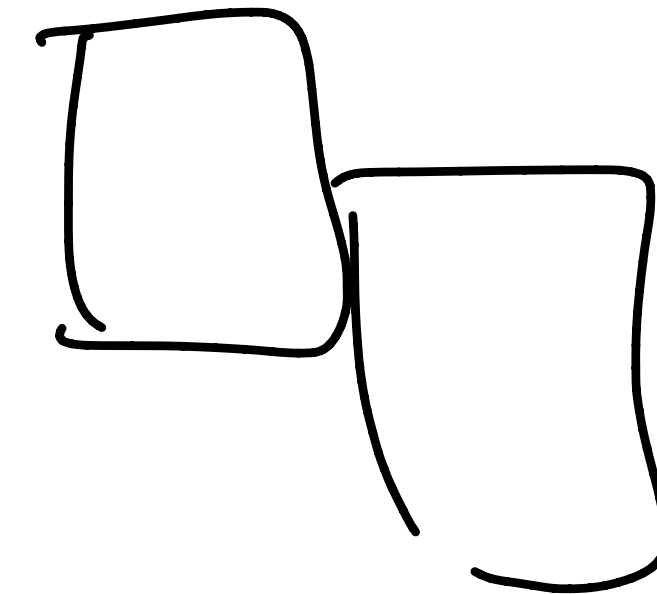
<https://www.acmicpc.net/problem/16937>

스티커 2개를 322

붙이는 문제

27

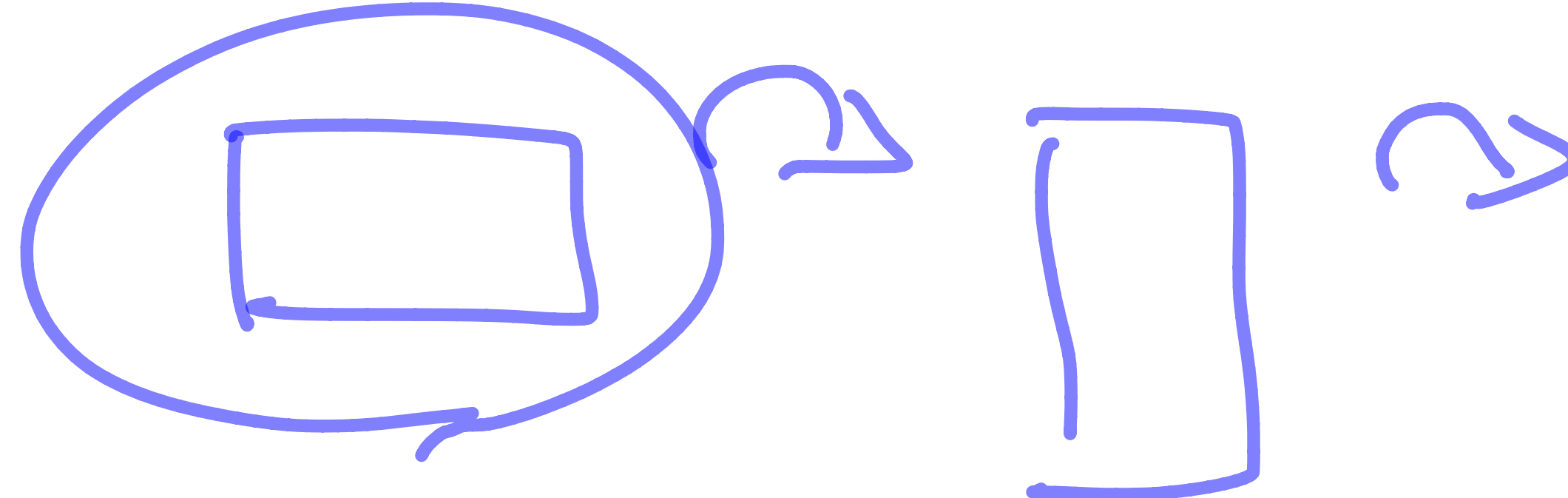
- 크기가 $H \times W$ 인 모눈 종이가 있고, 스티커 N 개가 있다. $1 \leq H, W, N \leq 100$
- i 번 스티커의 크기는 $R_i \times C_i$ 이다. 스티커는 회전시킬 수 있다.
- 스티커 2개를 붙이려고 한다. 두 스티커는 겹치면 안된다. 접하는 것은 가능하다.
- 붙여진 넓이의 최댓값을 구하는 문제



NC2

두 스티커

<https://www.acmicpc.net/problem/16937>



28

- 크기가 $H \times W$ 인 모눈 종이가 있고, 스티커 N 개가 있다. $1 \leq H, W, N \leq 100$
- i 번 스티커의 크기는 $R_i \times C_i$ 이다. 스티커는 회전시킬 수 있다. 90°
- **스티커 2개**를 붙이려고 한다. 두 스티커는 겹치면 안된다. 접하는 것은 가능하다.
- 붙여진 넓이의 최댓값을 구하는 문제
- 경우의 수 = N^2

두 스티커

<https://www.acmicpc.net/problem/16937>

- 크기가 $H \times W$ 인 모눈 종이가 있고, 스티커 N 개가 있다. $1 \leq H, W, N \leq 100$
- i 번 스티커의 크기는 $R_i \times C_i$ 이다. 스티커는 **회전**시킬 수 있다.
- **스티커 2개**를 붙이려고 한다. 두 스티커는 겹치면 안된다. 접하는 것은 가능하다.
- 붙여진 넓이의 최댓값을 구하는 문제
- 경우의 수 = $N^2 \times 2^2$

두 스티커

<https://www.acmicpc.net/problem/16937>

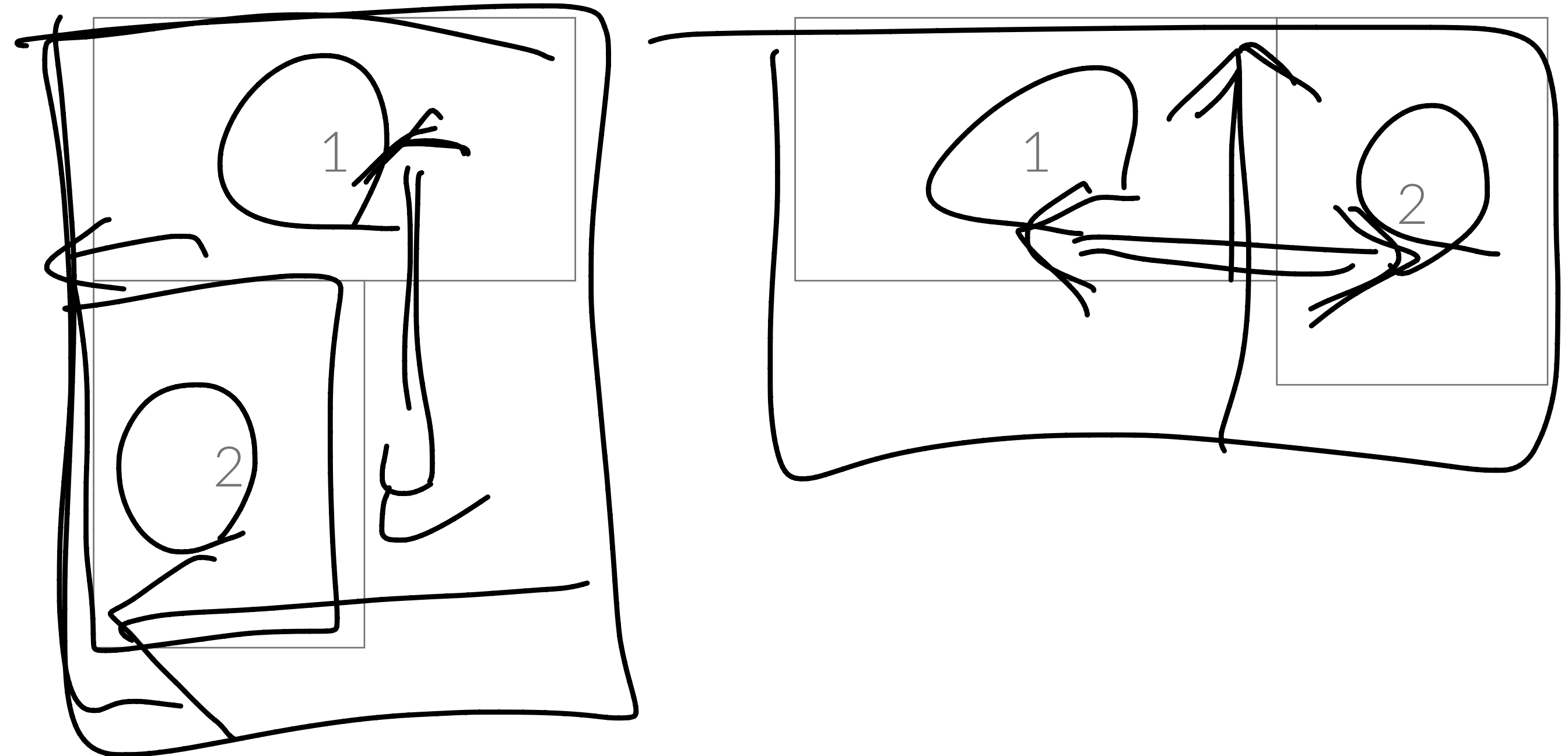
- 크기가 $H \times W$ 인 모눈 종이가 있고, 스티커 N 개가 있다. $1 \leq H, W, N \leq 100$
- i 번 스티커의 크기는 $R_i \times C_i$ 이다. 스티커는 회전시킬 수 있다.
- 스티커 2개를 붙이려고 한다. 두 스티커는 겹치면 안된다. 접하는 것은 가능하다.

- 붙여진 넓이의 최댓값을 구하는 문제

- 경우의 수 = $N^2 \times 2^2 \times 2$

$$8N^2$$

$$8 \times 100^2$$



두 스티커

<https://www.acmicpc.net/problem/16937>

- 소스: <http://codeplus.codes/44058ad5fb094d199dd90db0b81a1333>

캠프 준비

<https://www.acmicpc.net/problem/16938>

32

$$2^{15} = 32768$$

- 문제 N개의 난이도가 주어졌을 때 ($N \leq 15$)

2개 이상, $L \leq$ 난이도의 합 $\leq R$, 가장 어려운 문제의 난이도 - 가장 쉬운 문제의 난이도 $\geq X$

- 로 문제를 선택하는 방법의 수를 구하는 문제

1	2	...	N
0	0	...	0
x	x	...	x

$$2^N$$

캠프 준비

<https://www.acmicpc.net/problem/16938>

- 소스: <http://codeplus.codes/9c1292a845164107b74b0e00abe36b48>

캠프 준비

<https://www.acmicpc.net/problem/16938>

- 소스: <http://codeplus.codes/74c36abd7a39422f89e5b317314c141f>

숫자 재배치

<https://www.acmicpc.net/problem/16943>

35

91 = 362880711

- 두 정수 A, B가 있을 때, A에 포함된 숫자의 순서를 섞어서 새로운 수 C를 만든다.
- 가능한 C중에서 B보다 작거나 같으면서, 가장 큰 값을 구하는 문제
- C는 0으로 시작하면 안된다.
- $1 \leq A, B \leq 10^9$

10⁹

9자리

10자리

1 0 0 0 0 0 0 0 0 0

숫자 재배치

<https://www.acmicpc.net/problem/16943>

- 수의 길이가 9보다 작거나 같기 때문에, 모든 순열을 살펴보는 것이 가능하다.

숫자 재배치

<https://www.acmicpc.net/problem/16943>

- 소스: <http://codeplus.codes/bf253a98a57748ae99b8a911017e0db8>

숫자 재배치

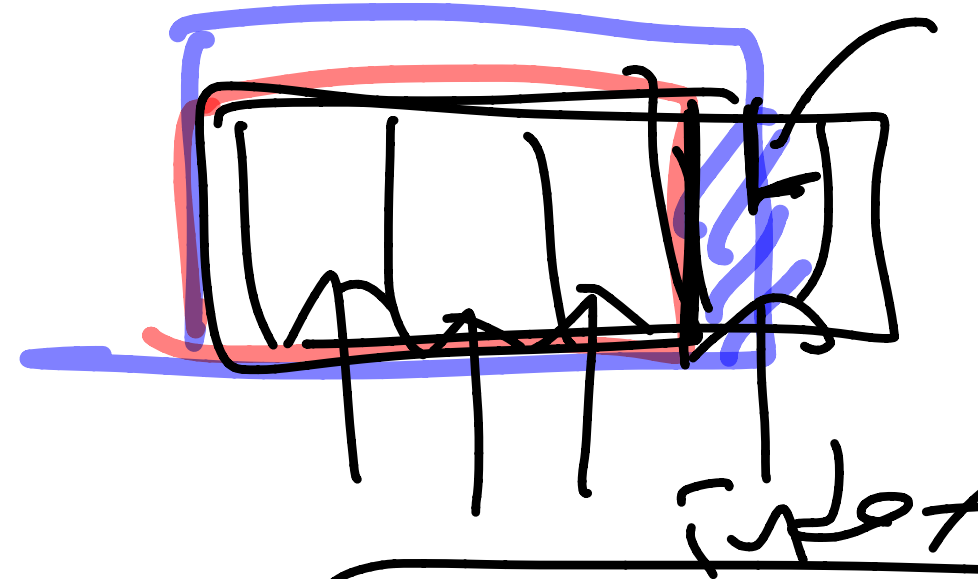
<https://www.acmicpc.net/problem/16943>

- 재귀 함수를 이용해서도 풀 수 있다.

숫자 재배치

39

<https://www.acmicpc.net/problem/16943>



```
int go(int index, int num) { // index 번째 수 추가, 현재까지 만든 수 = num
    if (index == n) return num;
    int ans = -1;
    for (int i=0; i<n; i++) {
        if (check[i] == true) continue; // 이미 사용한 수
        if (index == 0 && a[i] == 0) continue; // 0으로 시작하면 안됨
        check[i] = true;
        int temp = go(index+1, num*10+a[i]);
        if (temp < b) {
            if (ans == -1 || ans < temp) ans = max(ans, temp);
        }
        check[i] = false;
    }
    return ans;
}
```

숫자 재배치

<https://www.acmicpc.net/problem/16943>

재귀 $O(n)$

40

- 소스: <http://codeplus.codes/68e833fa2c5c4037ab4d1f76fd3a7bb8>

$$\boxed{3+8 \times 1} = 3+56 = 59$$

$$(3+8) \times 1 = 11 \times 1 = 11$$

괄호 추가하기

<https://www.acmicpc.net/problem/16637>

41

- 길이가 N 인 수식이 있다. ($N \leq 19$, 홀수)
- 수식의 정수는 0부터 9까지, 연산자는 $+$, $-$, \times 이다. 연산자 우선순위는 모두 같다.
- $3+8\times 7-9\times 2$ 의 결과는 136이다.
- 식에 괄호를 추가하면, 괄호 안에 들어있는 식은 먼저 계산해야 한다.
- 괄호 안에는 연산자가 하나만 들어 있어야 하고, 중첩된 괄호는 사용할 수 없다.
- 괄호를 적절히 추가해서 만들 수 있는 결과의 최댓값을 구하는 문제
- $3+8\times 7-9\times 2$ 의 경우 정답은 136

$3+8\times 7-9\times 2$

~~$(1+2\times 3)$~~

$(1+(2\times 3))$

괄호 추가하기

<https://www.acmicpc.net/problem/16637>

- 길이가 N인 수식이 있다. ($N \leq 19$, 홀수) 1차
- 수식의 정수는 0부터 9까지, 연산자는 $+$, $-$, \times 이다. 연산자 우선순위는 모두 같다.

- 연산자의 개수는 $(N-1)/2$ 개이다.

$$M = (N-1)/2$$

- 중첩된 괄호를 사용할 수 없고, 연산자 우선순위는 모두 같다.

$$M \leq (19-1)/2 = 9$$

- 따라서, 어떤 연산자를 먼저 계산할지 알면 된다.

2^9 경우의 수
512

괄호 추가하기

<https://www.acmicpc.net/problem/16637>

43

$$A+B =$$
$$A-B =$$

$$\frac{(A+B) + 0}{(A-B) + 0}$$

- $3+8 \times 7-9 \times 2$ 에 괄호를 $(3+8) \times (7-9) \times 2$ 로 친 것은 아래와 같이 변형해서 계산할 수 있다.

$(3+(8 \times 7))$ $A+B =$ $(A \times B) + 0$

3	+	8	×	7	-	9	×	2
		먼저	먼저					
11	+	0	×	-2	+	0	×	2

$(3+8 \times 7)$

$11 \times (-2) + 2 = -44$

-44

괄호 추가하기

<https://www.acmicpc.net/problem/16637>

- $3+8\times 7-9\times 2$ 에 괄호를 $3+(8\times 7)-(9\times 2)$ 로 친 것은 아래와 같이 변형해서 계산할 수 있다.

a =

b →

3	+	8	×	7	-	9	×	2
	○		○ 먼저		○		○ 먼저	
3	+	56	+	0	-	18	+	0

110 계산

괄호 추가하기

45

<https://www.acmicpc.net/problem/16637>

- 소스: <http://codeplus.codes/3832c89991b44007869612a89cc37b64>

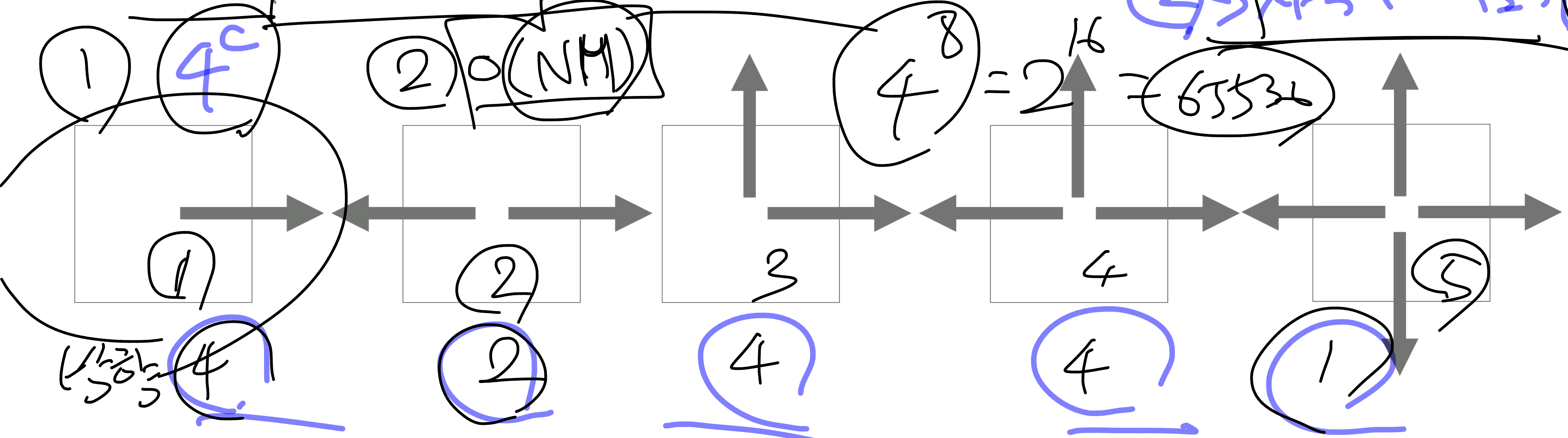
$$\textcircled{1} \times \textcircled{2}$$

- $N \times M$ 의 격자로 이루어진 사무실이 있다

(• 각 칸은 빈 칸, 벽, CCTV 중 하나

- CCTV는 총 5가지 종류가 있고, 방향을 정할 수 있다 (CCTV의 개수 ≤ 8)

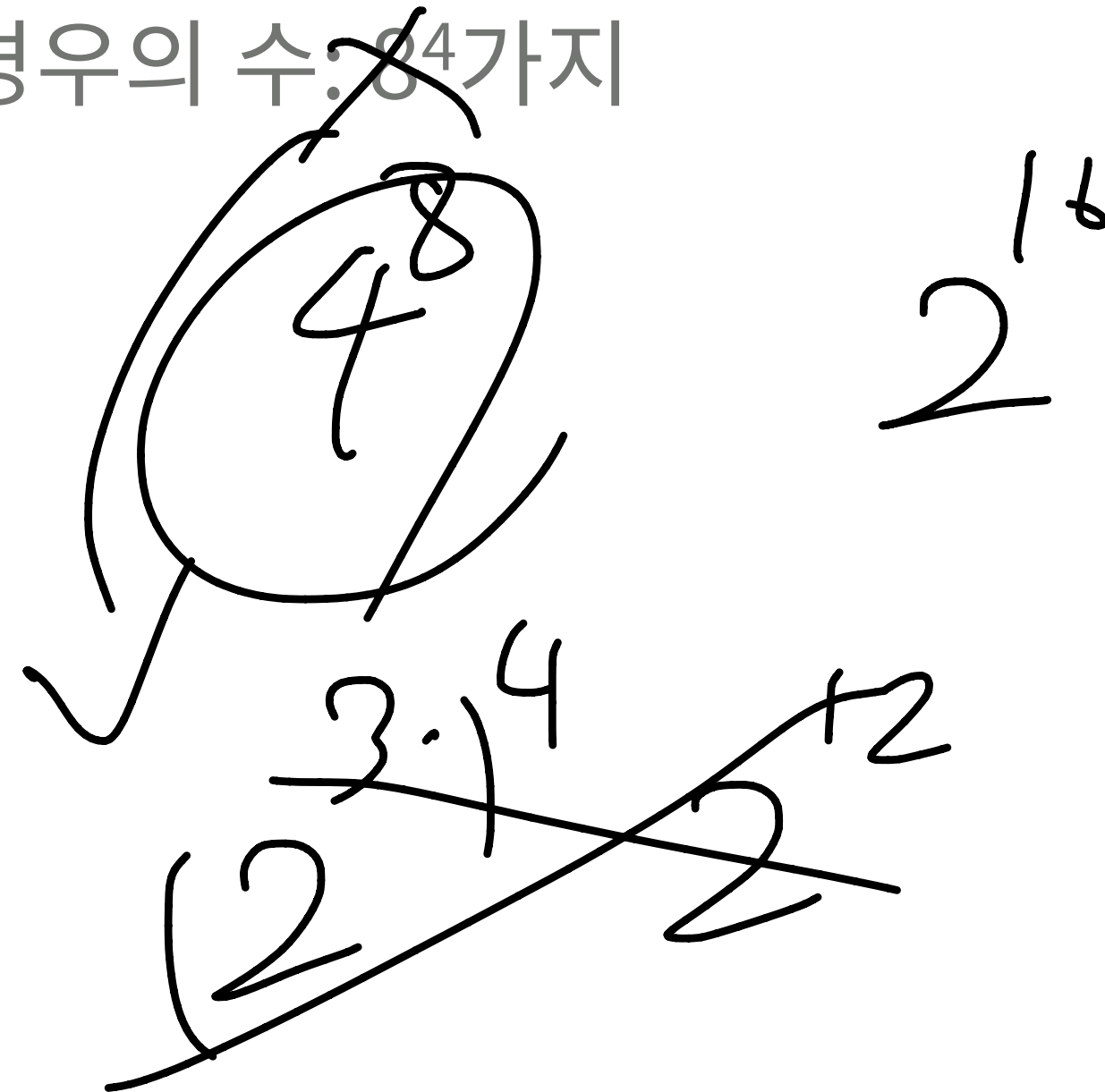
- 사각 지대 (감시받지 않는 영역)의 최소 크기를 구하는 문제



감시

<https://www.acmicpc.net/problem/15683>

- CCTV의 개수 ≤ 8 이고,
- 각 CCTV마다 가능한 방향은 4가지, 2가지, 4가지, 4가지, 1가지가 있다
- 편의상 4가지라고 가정해도, 총 가능한 경우의 수: ~~84~~가지



감시

48

<https://www.acmicpc.net/problem/15683>

- 소스: <http://codeplus.codes/ce0cef06d7964e8bb3fc2bd8e3660912>

등차수열 변환

<https://www.acmicpc.net/problem/17088>

49

- 길이가 N 인 수열 $A = A[1], A[2], \dots, A[N]$ 이 있을 때, 이 수열을 등차수열로 만드는 연산 횟수의 최솟값을 구하는 문제 $1 \leq N \leq 100,000$
- 연산 ± 1 을 더하거나 빼는 것, 각각의 수에는 연산을 1번만 사용할 수 있다.

$N \leq 100,000$

$A[2]$
 $A[2] - 1$
 $A[2] + 1$

$A[2]$ 에

$A[2]$
 $A[2] - 1$
 $A[2] + 1$

등차수열:

$$a_n = a_0 + (n-1)d$$

3
 a_2

a_2

$$d = a_1 - a_0$$

등차수열 변환

50

<https://www.acmicpc.net/problem/17088>

- 처음 두 개의 수가 정해지면, 등차수열을 만들 수 있다.
- 처음 두 개의 수를 정하는 방법은 총 $3 \times 3 = 9$ 가지가 있고, 이걸 모두 시도해보면 된다.

$$\begin{array}{r} 3 + 3 = 9 \\ \hline \end{array}$$

등차수열 변환

51

<https://www.acmicpc.net/problem/17088>

- 소스: <http://codeplus.codes/390c1369fd364b26bf682882a5fe5bb5>

치킨 배달

치킨

52

<https://www.acmicpc.net/problem/15686>

- $N \times N$ 크기의 도시가 있다. ($2 \leq N \leq 50$)

- 도시의 각 칸은 빈 칸, 치킨집, 집 중 하나

- 치킨 거리는 집과 가장 가까운 치킨집 사이의 거리

- 도시의 치킨 거리는 모든 집의 치킨 거리의 합

- 두 칸 $(r1, c1)$ 과 $(r2, c2)$ 사이의 거리는 $|r1-r2| + |c1-c2|$

- 도시의 치킨 거리가 가장 작아지게 치킨집 중에서 M 개를 고르는 문제 ($1 \leq M \leq 13$)

- $M \leq$ 치킨집의 개수 ≤ 13

13개 칸이 M개를 고르는 문제

1	2	-	-	-	13
0	0	-	-	-	0
X	X				X

$$2^{13} = 8 \times 1024 = 8192$$

치킨 배달

<https://www.acmicpc.net/problem/15686>

- 13개 중에서 M개를 고르는 문제이다

치킨 배달

<https://www.acmicpc.net/problem/15686>

- 13개 중에서 M개를 고르는 문제이다 -> 2^{13} 가지 경우의 수가 있다

치킨 배달

55

<https://www.acmicpc.net/problem/15686>

- 소스: <http://codeplus.codes/20a4beedddc441bb8a848606b3761450>

숫자판 점프

<https://www.acmicpc.net/problem/2210>

56

- 5×5 크기의 숫자판이 있다.
- 임의의 위치에서 시작해서, 인접해 있는 네 방향으로 다섯 번 이동하면서, 각 칸에 적혀있는 숫자를 차례로 붙인다.
- 이렇게 만들 수 있는 서로 다른 여섯 자리 수의 개수를 구하는 문제

$$25 \times \frac{16 \times 16 \times 16 \times 16 \times 16}{1024} = \text{약 } 250000$$

숫자판 점프

<https://www.acmicpc.net/problem/2210>

- 5×5 개의 칸에서 시작해서, 총 5번 인접한 4 칸으로 이동할 수 있다.

숫자판 점프

<https://www.acmicpc.net/problem/2210>

- 5×5 개의 칸에서 시작해서, 총 5번 인접한 4 칸으로 이동할 수 있다.
- 25×4^5

숫자판 점프

59

<https://www.acmicpc.net/problem/2210>

- `go(x, y, num, len)`

- 현재 (x, y)에 있고, 지금까지 만든 수의 길이는 len, 만든 수는 num

(x, y)

→ (nx, ny)

len+1

$\square \times 10 + \frac{1}{2}2$

x행, y열

1번 (2)

이동 횟수

숫자판 점프

60

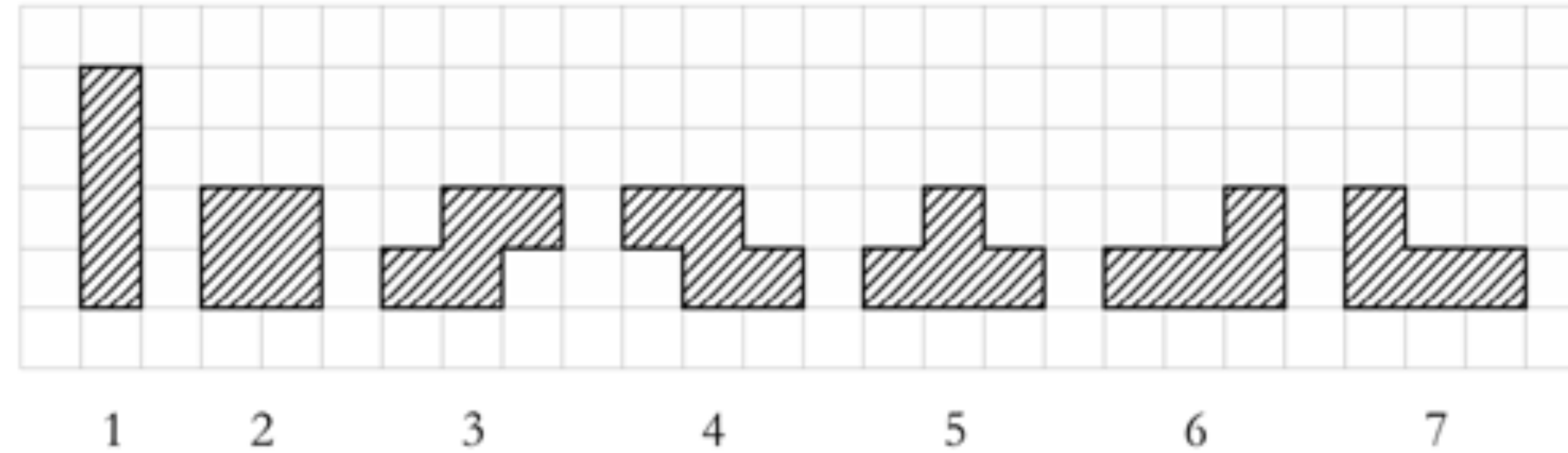
<https://www.acmicpc.net/problem/2210>

- 소스: <http://codeplus.codes/40d7537361484e2abf801d06a1dfa752>

테트리스

<https://www.acmicpc.net/problem/3019>

61



- 테트리스 블록은 오른쪽과 같다
- 블록이 떨어졌을 때, 블록과 블록 또는 블록과 바닥 사이에 채워져있지 않은 칸이 생기면 안된다
- 높이가 2, 1, 1, 1, 0, 1인 경우, 블록 5번을 떨어뜨리는 방법의 수는 5가지이다
- 테트리스 필드의 각 칸의 높이와
- 떨어뜨려야 하는 블록의 번호가 주어졌을 때
- 블록을 놓는 서로 다른 방법의 수를 구하는 문제



테트리스

<https://www.acmicpc.net/problem/3019>

- 블록을 떨어뜨리는 전체 경우의 수가 많지 않다
- 모든 경우를 시뮬레이션 해보면 된다

테트리스

<https://www.acmicpc.net/problem/3019>

- 소스: <http://codeplus.codes/6ca180c9c3f74e56a4057e71e2d2302b>

한윤정이 이탈리아에 가서...

64

<https://www.acmicpc.net/problem/2422>

$$N^3 = 2^3 = 8$$

- N종류의 아이스크림이 있고 ($1 \leq N \leq 200$)
- 3가지 아이스크림을 선택해서 먹으려고 한다
- 섞어먹으면 안되는 조합이 주어졌을 때 3가지 아이스크림을 선택하는 방법의 수
- 5개의 아이스크림과 3가지 섞어먹으면 안되는 조합이 있고, 1번은 2번 3번과 섞으면 안되고, 3번은 4번과 섞으면 안되는 경우 (1 4 5), (2 3 5), (2 4 5) 3가지 방법이 있다.

한윤정이 이탈리아에 가서...

65

<https://www.acmicpc.net/problem/2422>

- N이 작기 때문에, 모든 조합을 다 테스트해보면 된다

한윤정이 이탈리아에 가서...

66

<https://www.acmicpc.net/problem/2422>

```
int ans = 0;
for (int i=1; i<=n-2; i++) {
    for (int j=i+1; j<=n-1; j++) {
        for (int k=j+1; k<=n; k++) {
            if (a[i][j] || a[j][k] || a[k][i]) continue;
            ans += 1;
        }
    }
}
```

한윤정이 이탈리아에 가서...

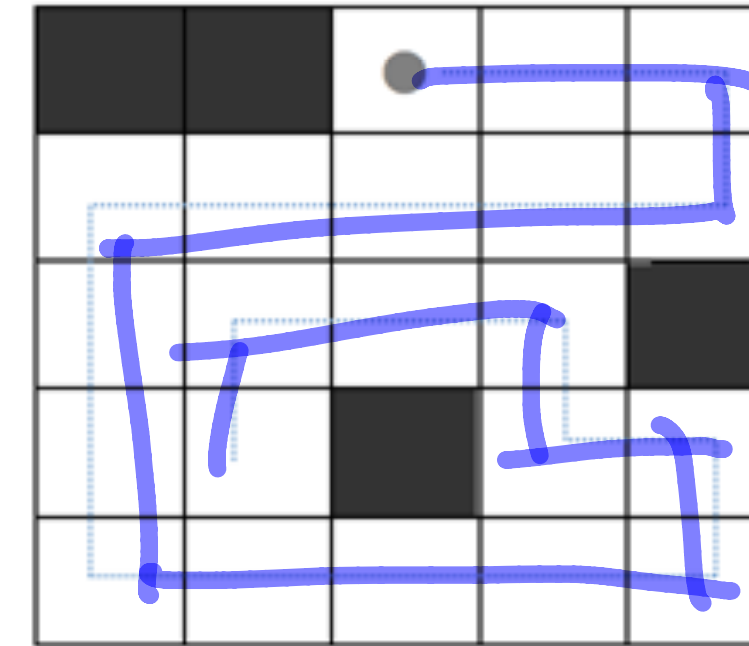
67

<https://www.acmicpc.net/problem/2422>

- 소스: <http://codeplus.codes/3a879bdec4a64b39a388a12930c3ea39>

NxM 보드 완주하기

<https://www.acmicpc.net/problem/9944>



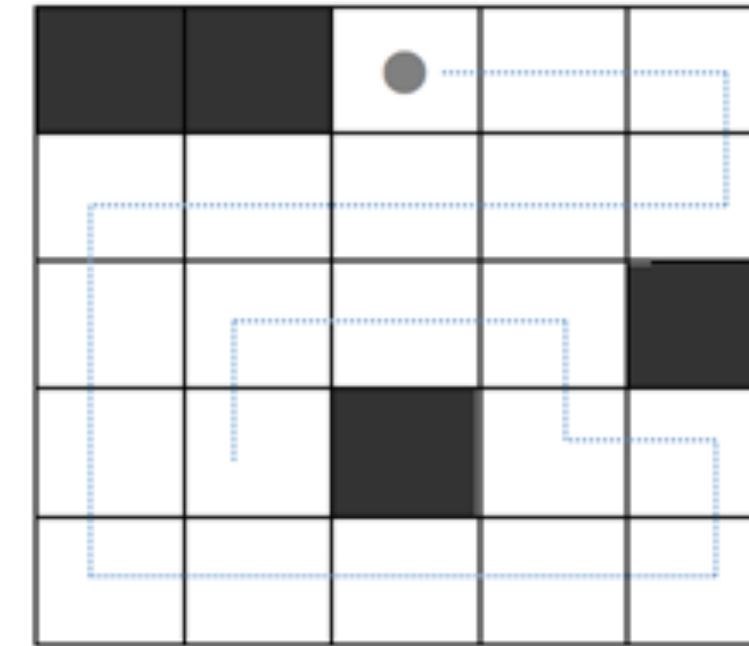
- NxM 보드가 있다. 보드의 각 칸은 빈 칸 또는 장애물이다.
- 보드 위에 공을 하나 놓고, 모든 칸을 방문하려고 한다.
- 공은 네 방향 중 한 방향으로 더 이상 이동하지 않을 때까지 이동한다.

장애물, 경계, 10/12/2/4칸칸

NxM 보드 완주하기

<https://www.acmicpc.net/problem/9944>

- 보드의 모든 칸 위에 공을 놓았다고 가정하고
- 모든 네 방향을 다 이동해본다.



NxM 보드 완주하기

70

<https://www.acmicpc.net/problem/9944>

- 소스: <http://codeplus.codes/676c4a5879244e64b2c9a78c4176f61f>

세 친구

<https://www.acmicpc.net/problem/17089>

N^3

4000^3

71

- N명의 사람 중에서 세 사람 A, B, C를 고르려고 한다. $3 \leq N \leq 4,000$
- 세 사람은 모두 친구여야 한다. (친구 관계 $\leq 4,000$)
- A의 친구 수 + B의 친구 수 + C의 친구 수가 최소가 되게 고르려고 한다.
- A의 친구 수를 구할 때는 B, C를 빼고 구한다. B, C의 친구를 구할 때도 마찬가지

64,000,000,000

6403

세 친구

<https://www.acmicpc.net/problem/17089>

- N명의 사람 중에서 세 사람 A, B, C를 고르려고 한다. $3 \leq N \leq 4,000$

• A, B를 구하는 방법 = $O(N^2)$

• 그 다음 C를 구하는 방법 = $O(N)$

• 총 $O(N^3)$

3

$i = 0 \sim n-1$
 $j = i+1 \sim n-1$
 $k = j+1 \sim n-1$

세 친구

<https://www.acmicpc.net/problem/17089>

- N명의 사람 중에서 세 사람 A, B, C를 고르려고 한다. $3 \leq N \leq 4,000$

A, B를 구하는 방법 = $O(N^2)$

A, B가 친구일 때만 C를 구한다.

그 다음 C를 구하는 방법 = $O(N)$

따라서, 총 $O(N^2 + MN)$

N^2

최대 M번만 수행

$N^2 + MN$

세 친구

74

<https://www.acmicpc.net/problem/17089>

- 소스: <http://codeplus.codes/3667996e11cd446dbe7af3aa160d606f>

배열 돌리기 4

<https://www.acmicpc.net/problem/17406>

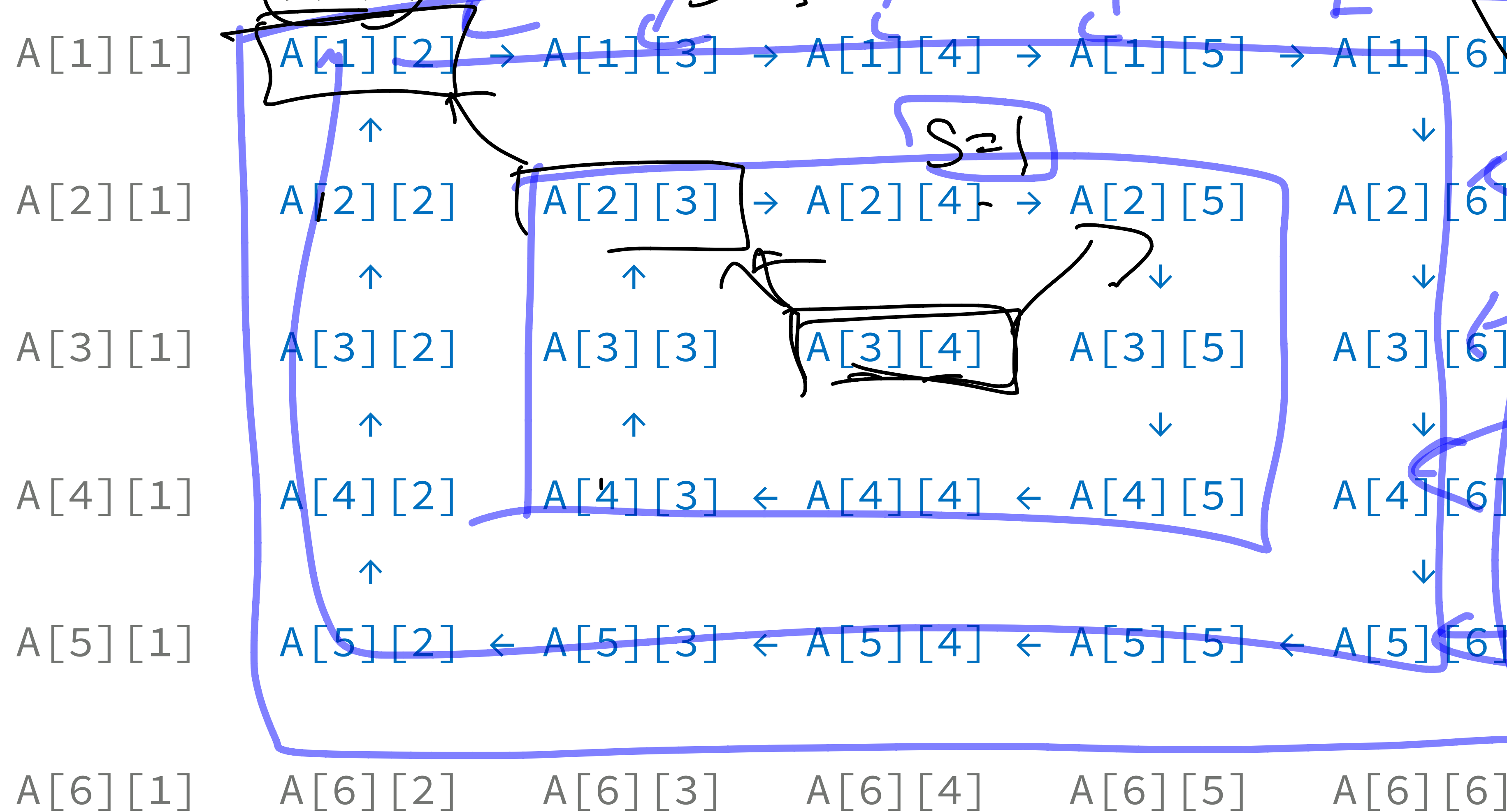
- 크기가 $N \times M$ 인 배열 A가 있고, 배열 A의 값은 행에 있는 모든 수의 합 중 최솟값이다.
- 회전 연산은 (r, c, s) 세 정수로 이루어져 있고, 가장 왼쪽 윗 칸이 $(r-s, c-s)$, 가장 오른쪽 아랫 칸이 $(r+s, c+s)$ 인 정사각형을 시계 방향으로 한 칸씩 돌리는 것이다.
- 회전 연산 K개의 순서를 정해서 배열 A의 값의 최솟값을 구하는 문제
- $N, M \leq 50, 1 \leq K \leq 6$

방법 1: $6! = 720$ 가지

배열 돌리기 4 $\begin{smallmatrix} 3 & 4 \\ 8 & 2 \end{smallmatrix}$ (3, 4)

<https://www.acmicpc.net/problem/17406>

- 회전 연산이 $(3, 4, 2)$ 인 경우


$$2\pi^0 \mid m^{\frac{1}{2}}_0$$

Handwritten notes:

1. $\frac{1}{2}$

2. $\frac{1}{2}$

3. $\frac{1}{2}$

4. $\frac{1}{2}$

5. $\frac{1}{2}$

6. $\frac{1}{2}$

7. $\frac{1}{2}$

8. $\frac{1}{2}$

9. $\frac{1}{2}$

10. $\frac{1}{2}$

11. $\frac{1}{2}$

12. $\frac{1}{2}$

13. $\frac{1}{2}$

14. $\frac{1}{2}$

15. $\frac{1}{2}$

16. $\frac{1}{2}$

17. $\frac{1}{2}$

18. $\frac{1}{2}$

19. $\frac{1}{2}$

20. $\frac{1}{2}$

21. $\frac{1}{2}$

22. $\frac{1}{2}$

23. $\frac{1}{2}$

24. $\frac{1}{2}$

25. $\frac{1}{2}$

26. $\frac{1}{2}$

27. $\frac{1}{2}$

28. $\frac{1}{2}$

29. $\frac{1}{2}$

30. $\frac{1}{2}$

31. $\frac{1}{2}$

32. $\frac{1}{2}$

33. $\frac{1}{2}$

34. $\frac{1}{2}$

35. $\frac{1}{2}$

36. $\frac{1}{2}$

37. $\frac{1}{2}$

38. $\frac{1}{2}$

39. $\frac{1}{2}$

40. $\frac{1}{2}$

41. $\frac{1}{2}$

42. $\frac{1}{2}$

43. $\frac{1}{2}$

44. $\frac{1}{2}$

45. $\frac{1}{2}$

46. $\frac{1}{2}$

47. $\frac{1}{2}$

48. $\frac{1}{2}$

49. $\frac{1}{2}$

50. $\frac{1}{2}$

51. $\frac{1}{2}$

52. $\frac{1}{2}$

53. $\frac{1}{2}$

54. $\frac{1}{2}$

55. $\frac{1}{2}$

56. $\frac{1}{2}$

57. $\frac{1}{2}$

58. $\frac{1}{2}$

59. $\frac{1}{2}$

60. $\frac{1}{2}$

61. $\frac{1}{2}$

62. $\frac{1}{2}$

63. $\frac{1}{2}$

64. $\frac{1}{2}$

65. $\frac{1}{2}$

66. $\frac{1}{2}$

67. $\frac{1}{2}$

68. $\frac{1}{2}$

69. $\frac{1}{2}$

70. $\frac{1}{2}$

71. $\frac{1}{2}$

72. $\frac{1}{2}$

73. $\frac{1}{2}$

74. $\frac{1}{2}$

75. $\frac{1}{2}$

76. $\frac{1}{2}$

77. $\frac{1}{2}$

78. $\frac{1}{2}$

79. $\frac{1}{2}$

80. $\frac{1}{2}$

81. $\frac{1}{2}$

82. $\frac{1}{2}$

83. $\frac{1}{2}$

84. $\frac{1}{2}$

85. $\frac{1}{2}$

86. $\frac{1}{2}$

87. $\frac{1}{2}$

88. $\frac{1}{2}$

89. $\frac{1}{2}$

90. $\frac{1}{2}$

91. $\frac{1}{2}$

92. $\frac{1}{2}$

93. $\frac{1}{2}$

94. $\frac{1}{2}$

95. $\frac{1}{2}$


96. $\frac{1}{2}$

97. $\frac{1}{2}$

98. $\frac{1}{2}$

99. $\frac{1}{2}$

100. $\frac{1}{2}$



271 245

$$37\frac{21}{2}$$

284가 정

$$\overline{(19)_7 : 2S+1}$$

배열 돌리기 4

<https://www.acmicpc.net/problem/17406>

- 가능한 순서가 $K! \leq 6! = 720$ 가지 밖에 안된다.
- 모든 순서를 다 만들고 해본다.

배열 돌리기 4

<https://www.acmicpc.net/problem/17406>

- 소스: <http://codeplus.codes/32a3c8caa6e8477fa7162743e39f1e26>