

# 카카오 모의테스트

최백준 [choi@startlink.io](mailto:choi@startlink.io)

---

# 문제 1

<http://startl.in/2vPxF8j>

2

$$\textcircled{724} \textcircled{123} = 6$$

- 자연수  $N$ 이 주어지면,  $N$ 의 각 자릿수의 합을 구하는 문제
- $N \leq 100,000,000$

$$7008 = 15$$

정답  $\rightarrow$  풀라

# 문제 1

<http://startl.in/2vPxF8j>

- 정수를 문자열로 바꿔서 각 자릿수의 합을 구하는 방법이 있다

# 문제 1

<http://startl.in/2vPx8j>

- C: `sprintf(str, "%d", num);`
- C++: `string str = to_string(num);`
- Java: `String str = Integer.toString(num);`

# 문제 1

<http://startl.ink/2vPxF8j>

- C++: <https://gist.github.com/Baekjoon/a131b00003f4749dab9a96b4f8f62cce>
- C++: <https://gist.github.com/Baekjoon/d84cb26739a8b598dc303acf7100a5ee>
- Java: <https://gist.github.com/Baekjoon/3c4e9104248d8c6b0547e6be6642777e>

# 문제 1

<http://startl.in/2vPx8j>

- 정수를 계속해서 10으로 나누면서 나머지를 더해나가는 것도 된다

$7056 \rightarrow \%10 \quad 6$   
 $\downarrow /10$   
 $705 \rightarrow \%10 \quad 5$   
 $\downarrow /10$   
 $70$

# 문제 1

<http://startl.ink/2vPxF8j>

- C++: <https://gist.github.com/Baekjoon/d969f4c643ce638889fb541114d34418>
- Java: <https://gist.github.com/Baekjoon/9deb1dd33cc5bc90d959a5019a8e9720>

# 문제 2

<http://startl.ink/2vOzp1E>

- 길이가 N인 배열이 주어졌을 때, 1부터 N까지의 원소가 빠짐없이 있는지 존재하는 문제

→ 1부터 N까지의 원소가 빠짐없이 있는지



# 문제 2

<http://startl.in/2vOzp1E>

- $\text{cnt}[i]$  = 배열 A에 들어있는  $i$ 의 개수
- cnt배열의 1부터 N까지 값이 1인지 확인하면 된다

$\text{Cnt}[i] == 1$   
시간복잡도:  $O(N)$   
공간:  $O(N)$

# 문제 2

<http://startl.ink/2vOzp1E>

- C++: <https://gist.github.com/Baekjoon/6cbe83806016dfb86c55f13210f02de3>
- Java: <https://gist.github.com/Baekjoon/408bc3a38bb21de086be39189a200431>

$i$	0	1	2	3	...	$N-1$
$A[i]$	1	2	3	4	...	$N$

$$A[i] == (i+1)$$

$$\sum_{i=1}^N \frac{1}{i} = N$$

배열 A

배열 A의 정렬

$$\frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \dots + \frac{1}{N}$$

정렬

$$N \lg N$$

# 문제 2

<http://startl.ink/2vOzp1E>

- 같은 수를 xor 연산하면 0이 된다

$$\begin{array}{r} 000 \\ 011 \\ 101 \\ 110 \\ \hline \end{array}$$

$$\underline{3} \text{ xor } \underline{3} = 0$$

$$\begin{array}{r} 101 \\ 011 \\ \hline \end{array}$$

# 문제 2

<http://startl.in/2vOzp1E>

$$\begin{aligned} & (1 \text{ xor } 2 \text{ xor } \dots \text{ xor } N) \\ & \text{xor } (\underbrace{A[0]}_2 \text{ xor } \underbrace{A[1]}_4 \text{ xor } \dots \text{ xor } A[N-1]) \end{aligned}$$

12

- 같은 수를 xor 연산하면 0이 된다
- 이 점을 이용해서 1부터 N까지의 수를 모두 xor한 결과와 배열 A에 들어있는 수를 xor한 결과가 0이면 모든 수가 있는 것이다

# 문제 2

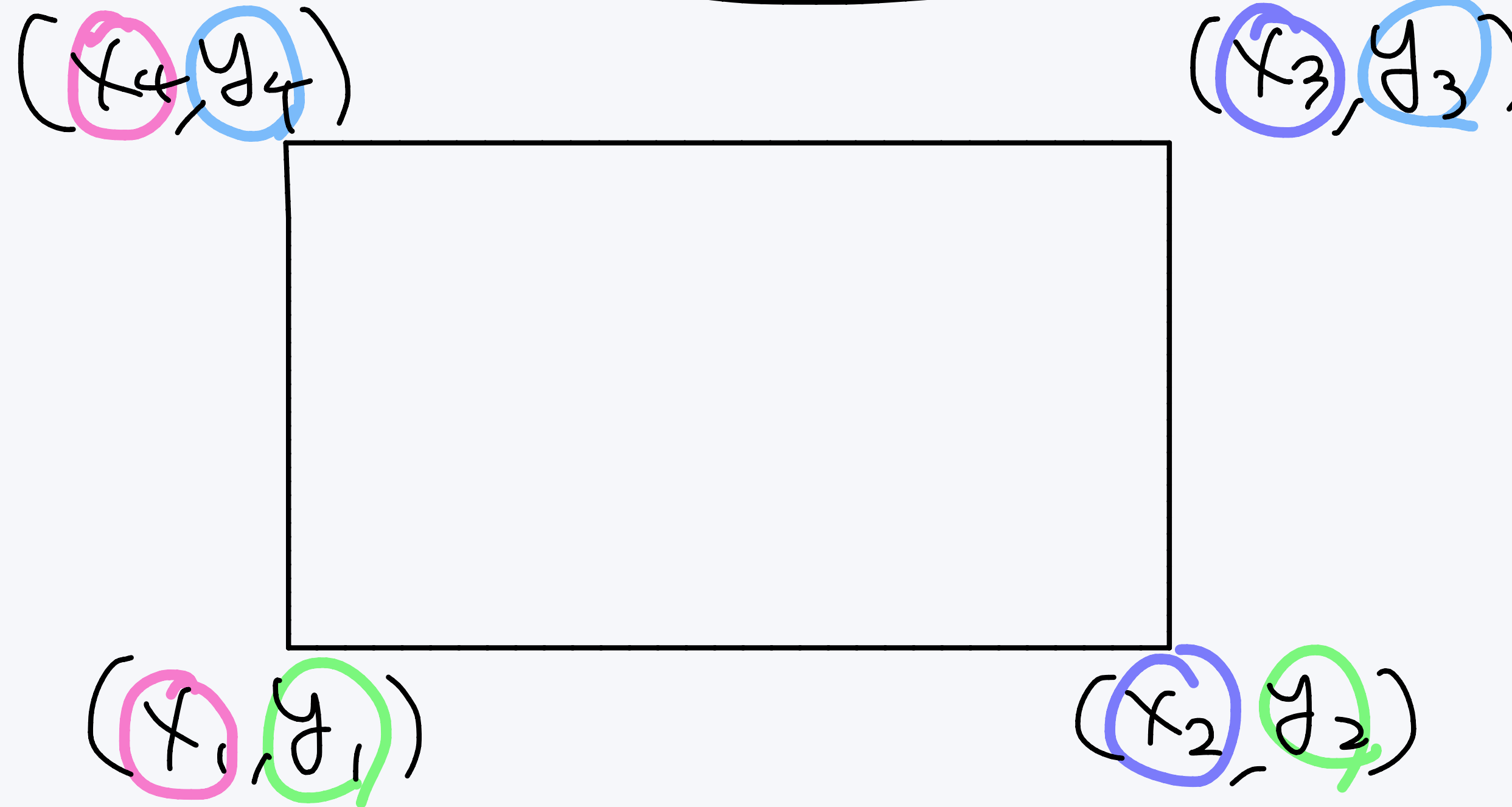
<http://startl.ink/2vOzp1E>

- C++: <https://gist.github.com/Baekjoon/9626fc84ff8c51411be1458acf497026>
- Java: <https://gist.github.com/Baekjoon/8f81b395dbe96cad039178c55296483a>

# 문제 3

<http://startl.in/2vPyT3x>

- 직사각형의 4점 중 3점이 주어졌을 때, 한 점을 구하는 문제
- 직사각형의 각 변은  $x, y$ 축에 평행하며, 항상 직사각형을 만들 수 있다



# 문제 3

<http://startl.ink/2vPyT3x>

15

- 입력으로 주어진 x좌표는 항상 2개가 같고, 1개가 다르다
- 입력으로 주어진 y좌표는 항상 2개가 같고, 1개가 다르다
- 다른 1개를 찾으면 된다

$$x_1 == x_2 ==$$

$$\frac{0+1+2}{\uparrow \uparrow \uparrow} = 3$$

$0 \quad 1 \quad 2$

$$\begin{aligned} (x_1) &== (x_2) \\ x_1 &== x_3 \\ x_2 &== x_3 \end{aligned}$$

$$\rightarrow x_3$$

$$\rightarrow x_2$$

$$\rightarrow x_1$$

$$3 - 1 - 2$$

# 문제 3

<http://startl.ink/2vPyT3x>

- C++: <https://gist.github.com/Baekjoon/e3019949ecbeef801f5694bb1062509a>
- Java: <https://gist.github.com/Baekjoon/70517a18af788f59799fde5d97944329>



# 문제 4

<http://startl.ink/2vPyzSo>

DP 다이내믹 프로그래밍

17

- 0과 1로 이루어진 표가 있을 때, 1로 이루어진 가장 큰 정사각형을 찾는 문제



# 문제 4

18

<http://startl.in/2vPyzSo>

- $D[i][j]$  =  $(i, j)$ 를 오른쪽 아래로 하는 가장 큰 정사각형 변의 길이

# 문제 4

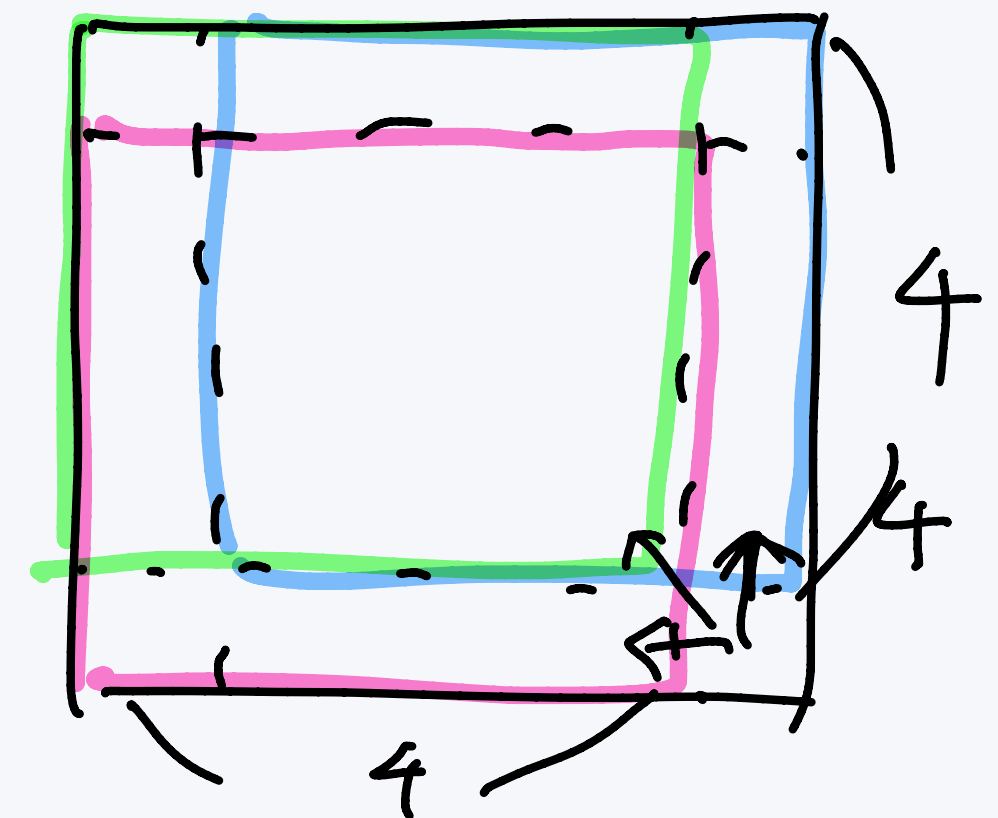
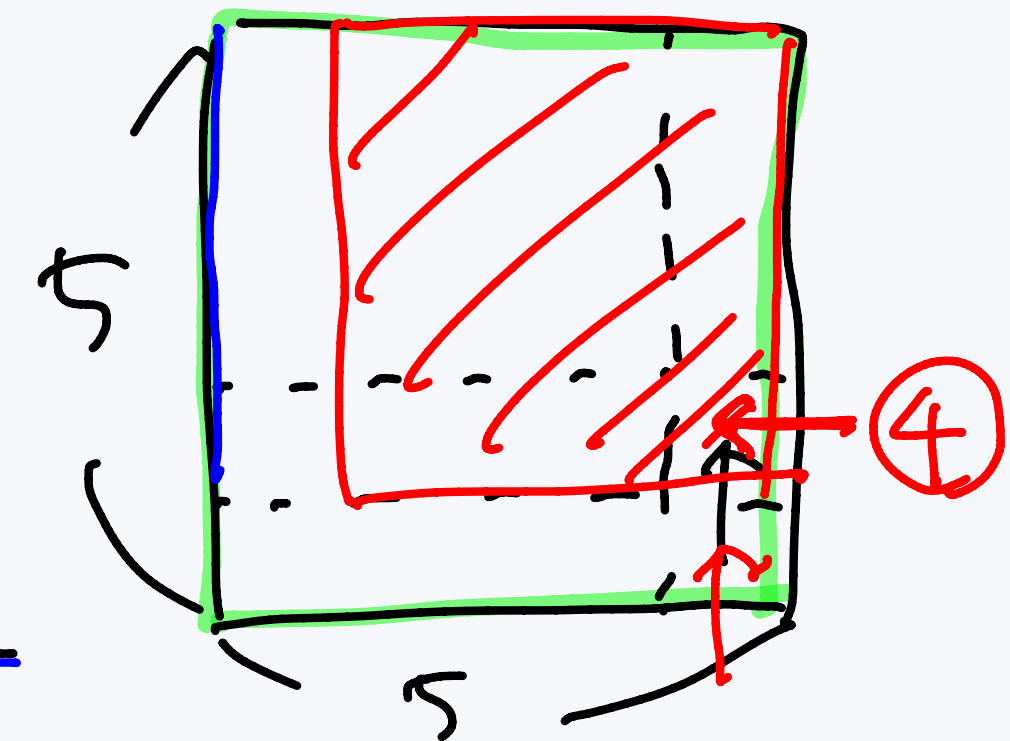
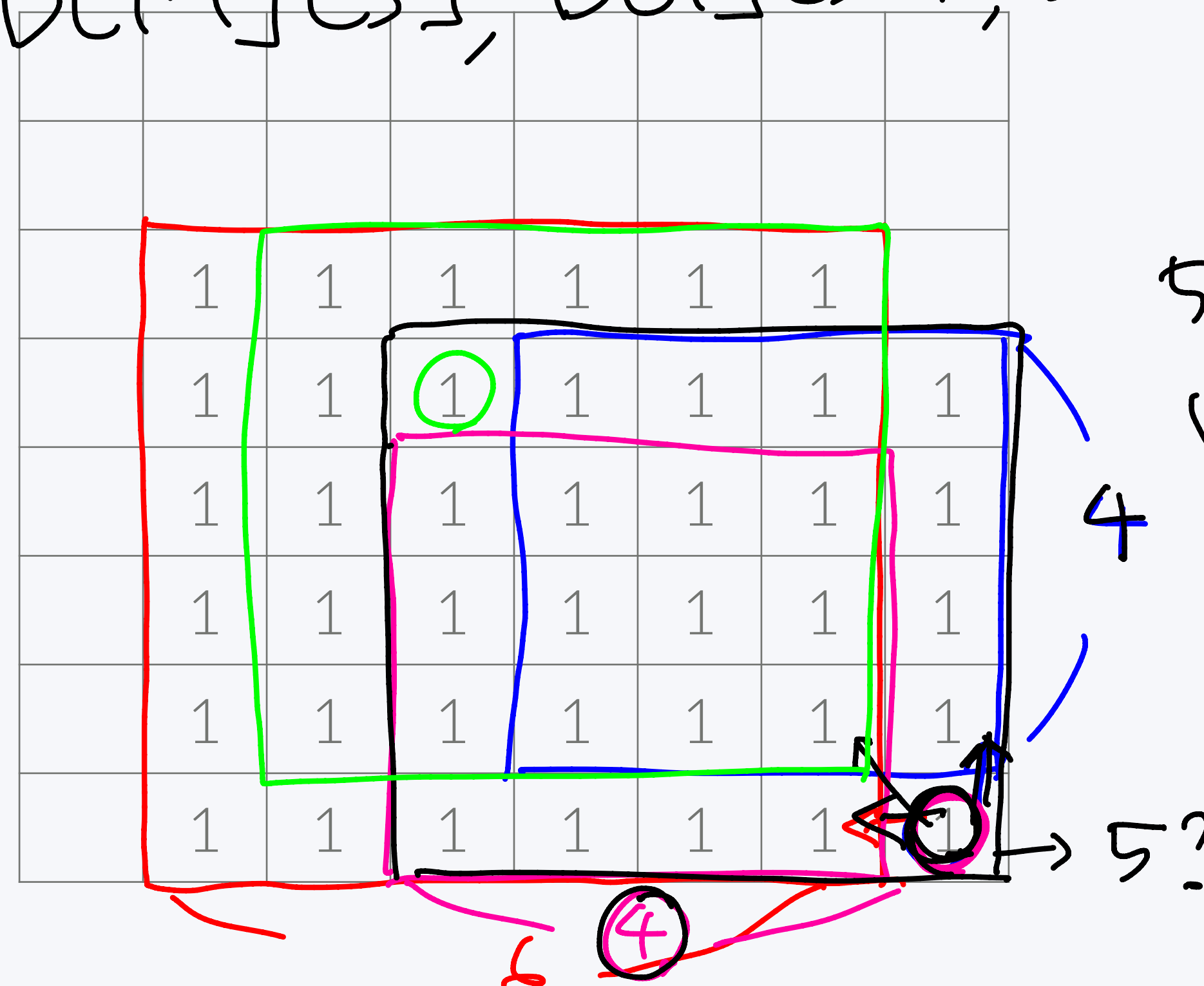
$N \leq 1,000$

$O(N^2)$

<http://startl.in/2vPyzSo>

- $D[i][j]$  = (i, j)를 오른쪽 아래로 하는 가장 큰 정사각형 변의 길이

$$D[i][j] = \min(D[i-1][j], D[i][j-1], D[i-1][j-1]) + 1$$

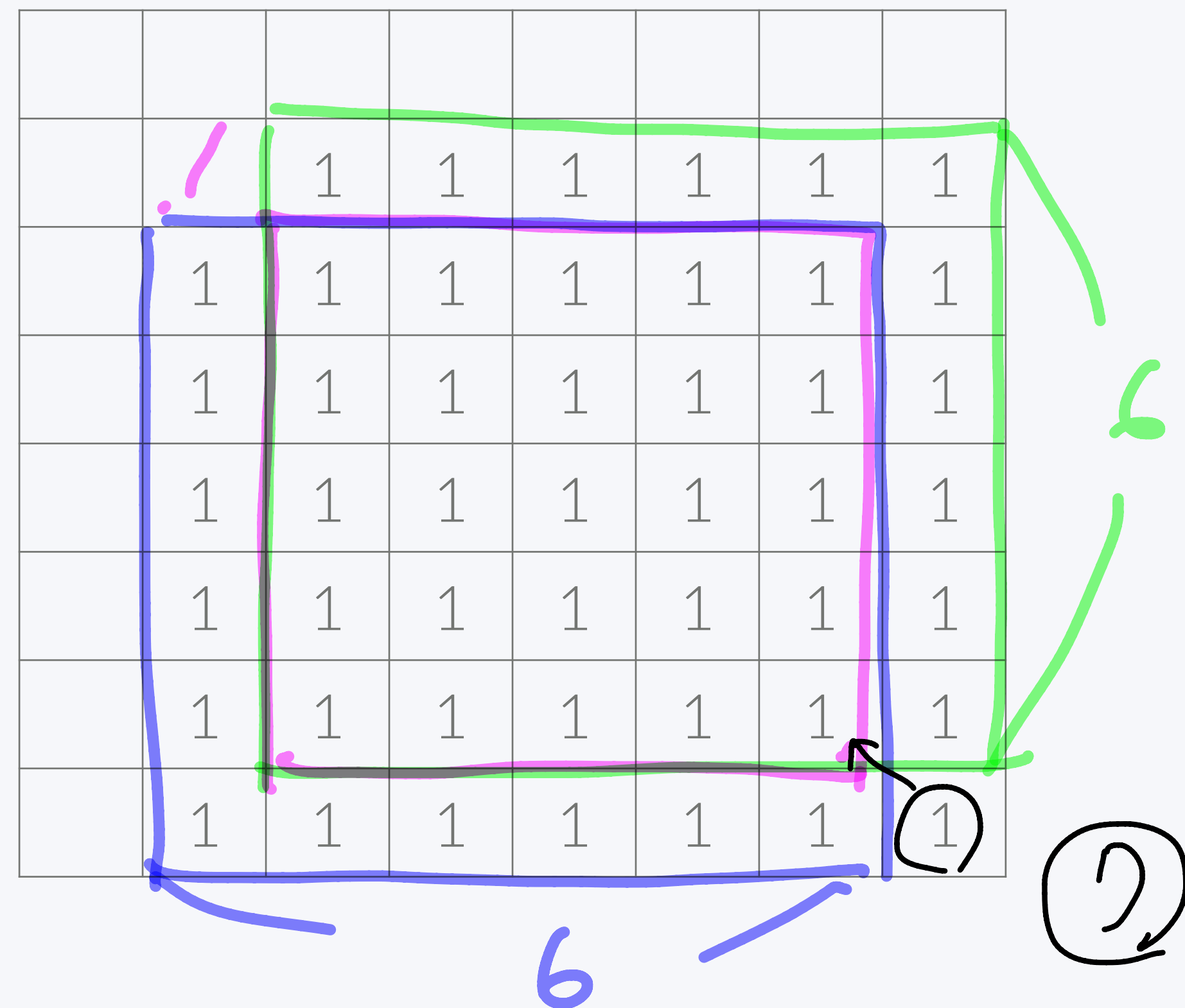


# 문제 4

20

<http://startl.ink/2vPyzSo>

- $D[i][j] = (i, j)$ 를 오른쪽 아래로 하는 가장 큰 정사각형 변의 길이



# 문제 4

21

<http://startl.ink/2vPyzSo>

- $D[i][j] = (i, j)$ 를 오른쪽 아래로 하는 가장 큰 정사각형 변의 길이

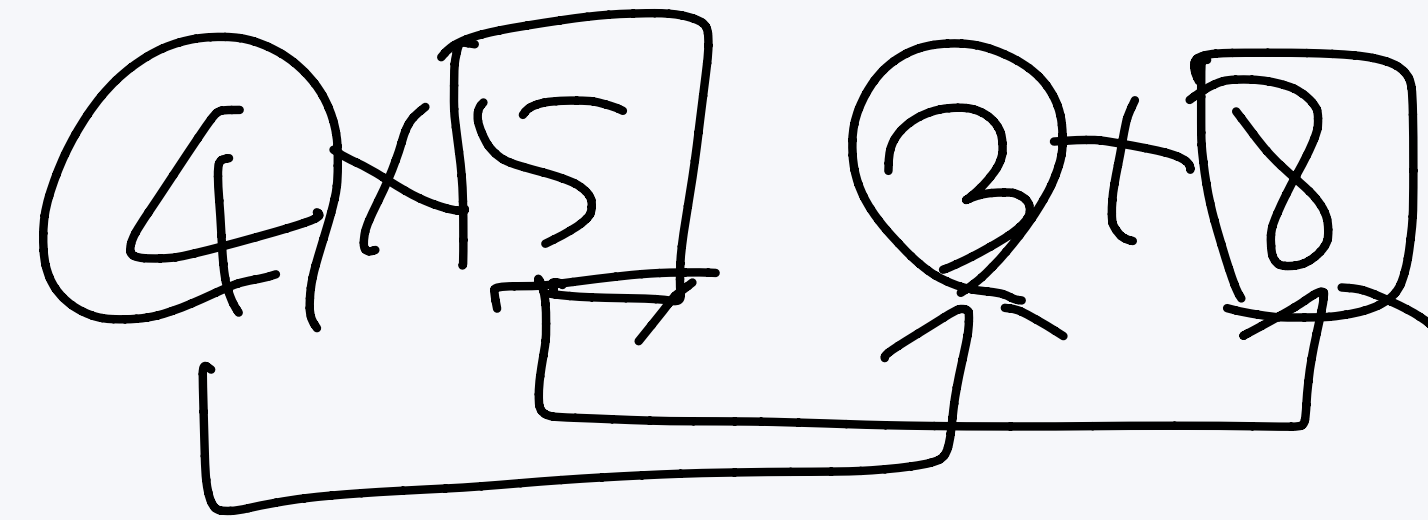
0	1	0	0	0	0	0	0
0	1	0	1	1	1	1	0
1	1	0	1	1	1	1	0
1	1	1	1	1	1	1	1
→ 0	0	0	0	0	1	0	1
→ 1	1	1	1	1	1	1	1
→ 1	1	0	0	0	1	1	1
→ 1	1	0	0	0	1	1	1
	1	1	1	1	1	1	1
	1	1	1	1	1	1	1

$$O(N) \times N$$

$$O(N^2)$$

# 문제 4

<http://startl.ink/2vPyzSo>



- $D[i][j] = (i, j)$ 를 오른쪽 아래로 하는 가장 큰 정사각형 변의 길이

각각의 칸을 가장 오른쪽  
높이가 1인 직사각형

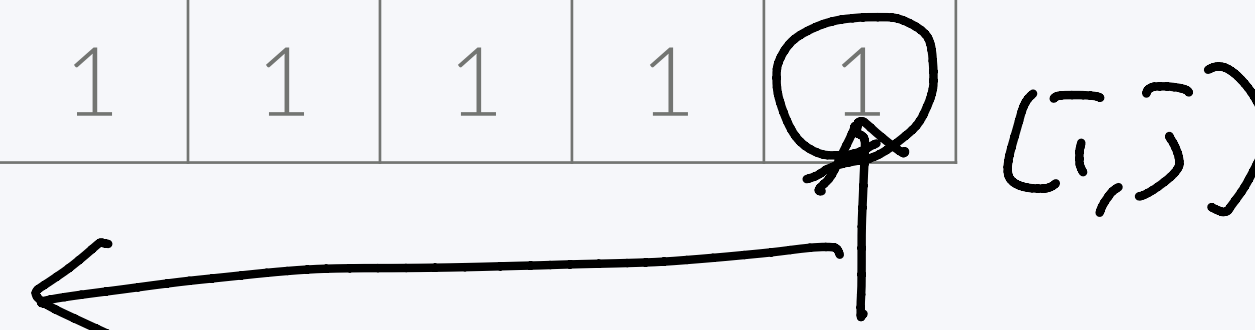
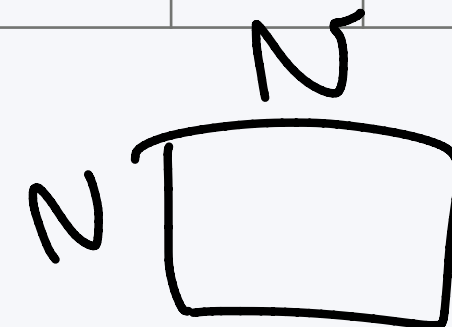
A 1 1 0 0 0 1 1 1 0 1  
D 1 2 0 0 0 1 2 3 4 0 1

1 1 1 1 1 1 1 1 1 1  
0 0 0 0 0 1 0 1 0 1  
1 1 1 1 1 1 1 1 1 1  
1 1 0 0 0 1 1 1 1 1  
1 1 0 0 0 1 1 1 1 1

9 5x2  
2 4x2  
9 3x4  
4 2x4  
4 1x4

시간복잡도:  $N^2 \times N$   
 $O(N^3)$

		1	1	1	1	1	1
		1	1	1	1	1	1
		1	1	1	1	1	1
		1	1	1	1	1	1
		1	1	1	1	1	1
		1	1	1	1	1	1
			1	1	1	1	1



# 문제 4

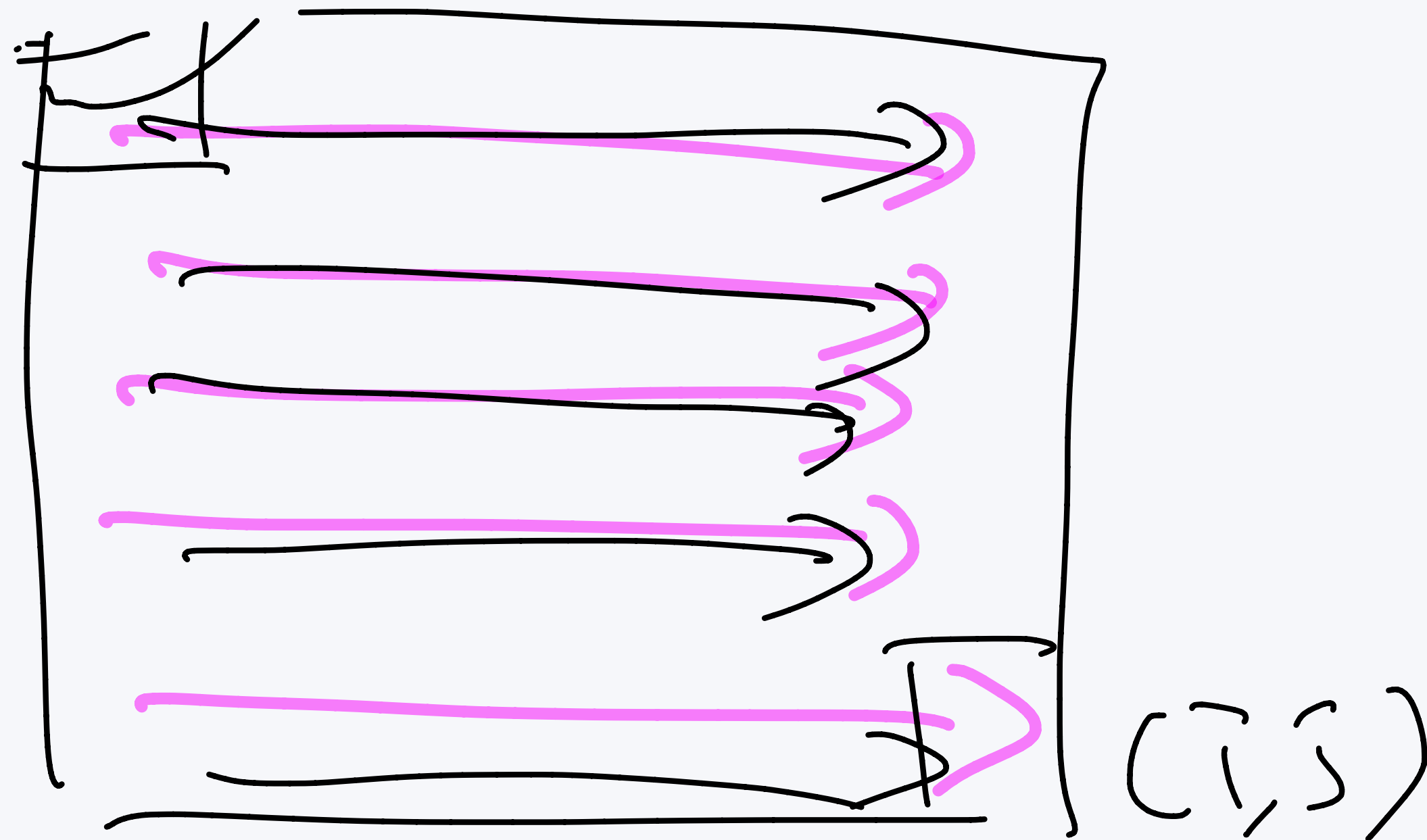
<http://startl.ink/2vPyzSo>

$N \times N$

가장 큰 정사각형 변의 길이:  $N^2$

가장 큰 정사각형

- $D[i][j] = (i, j)$ 를 오른쪽 아래로 하는 가장 큰 정사각형 변의 길이
- $D[i][j] = \min(D[i-1][j], D[i][j-1], D[i-1][j-1]) + 1$   $(i-1, j-1)$



$D[n][n]$  X

# 문제 4

<http://startl.ink/2vPyzSo>

- C++: <https://gist.github.com/Baekjoon/afdebccf12a45173670bc7916a4b492d>
- Java: <https://gist.github.com/Baekjoon/16a31015a58debecccdfe05be5a0a153>

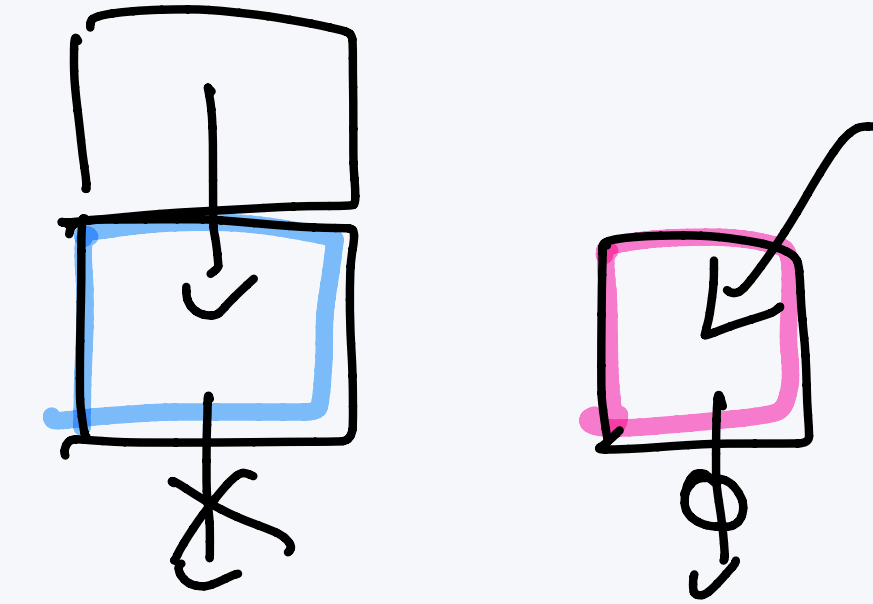


# 문제 5

<http://startl.ink/2vPMxDC>

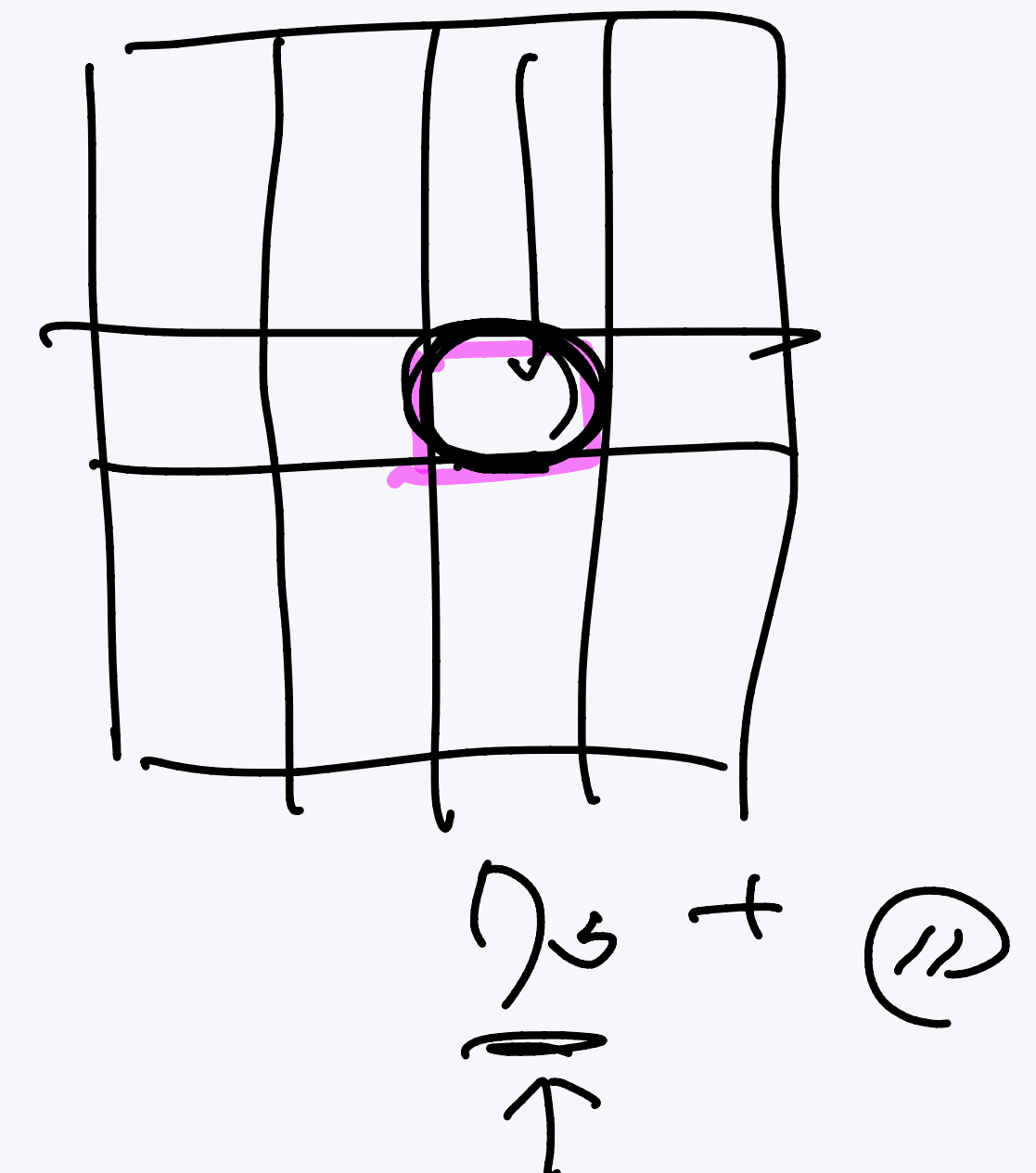
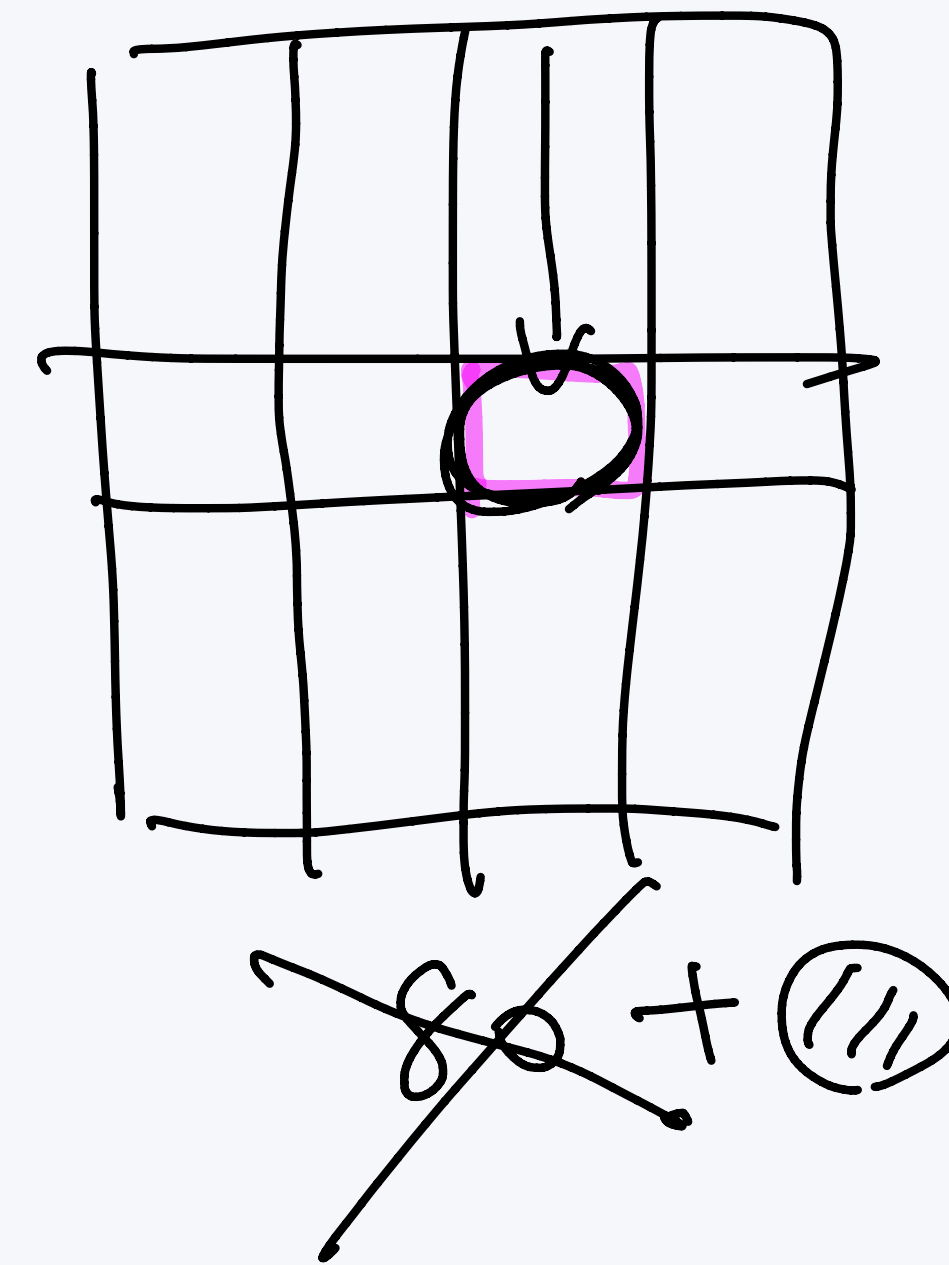
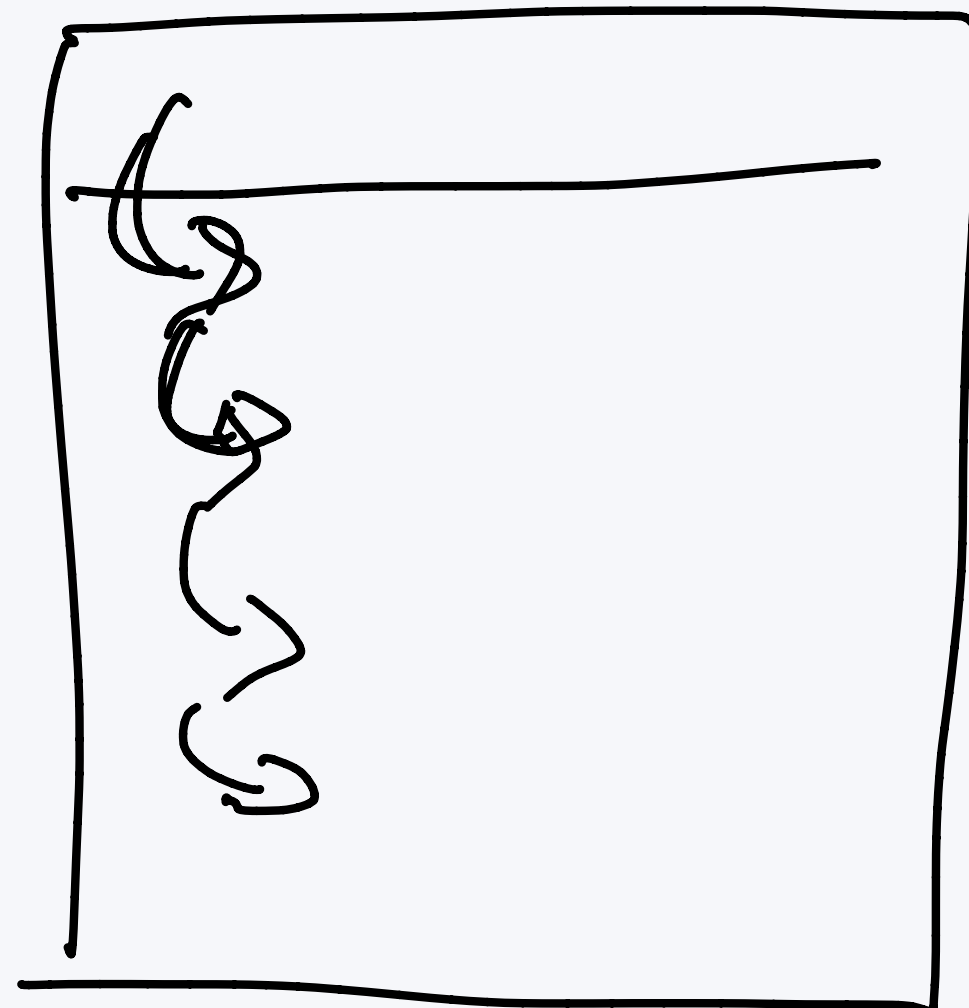
25

연속 3



- N행 4열로 이루어진 땅이 있고, 모든 칸에는 정수가 하나 쓰여져 있다
- 1행부터 땅을 밟아서 N행까지 내려와야 하고
- 각 행에서는 한 칸만 밟아야 한다
- 연속된 두 열을 밟을 수 없다

ex

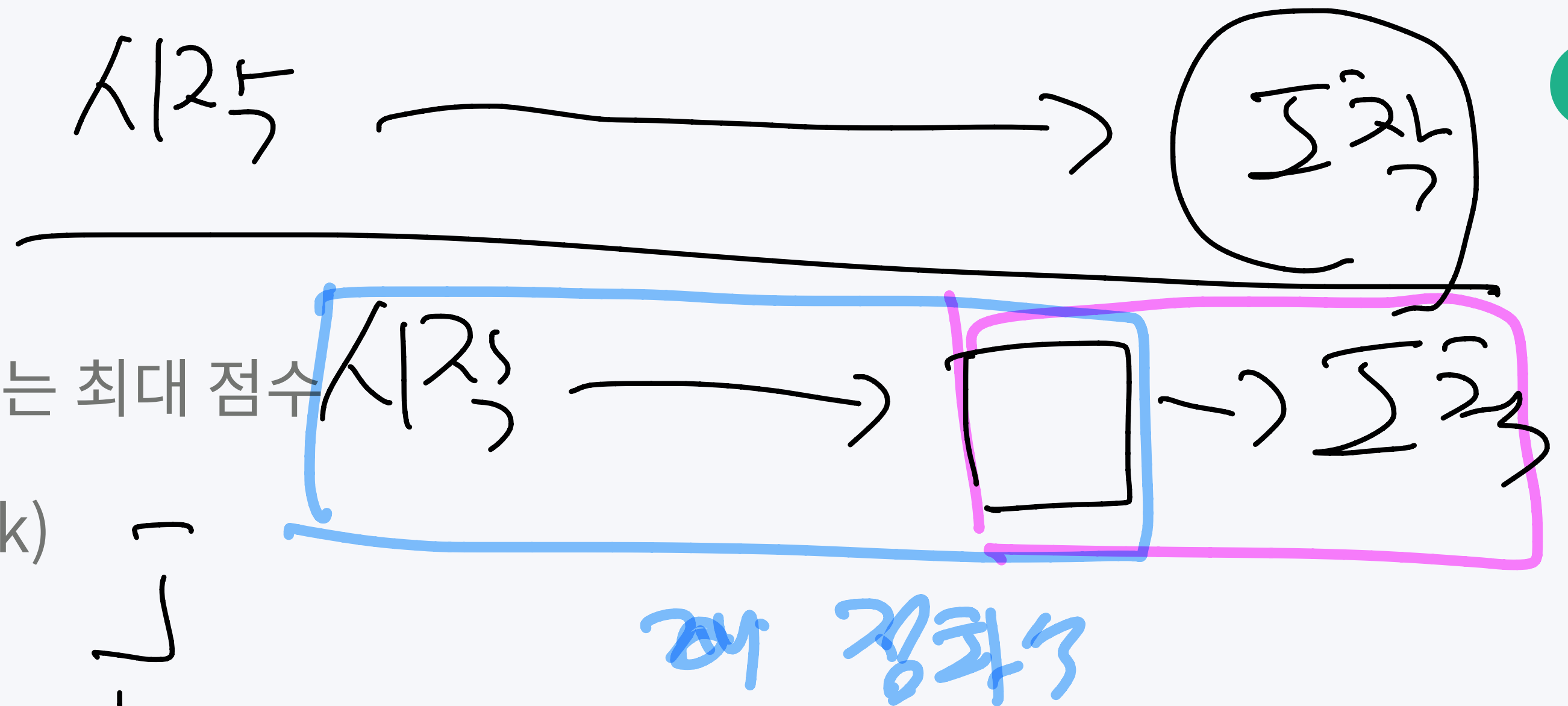
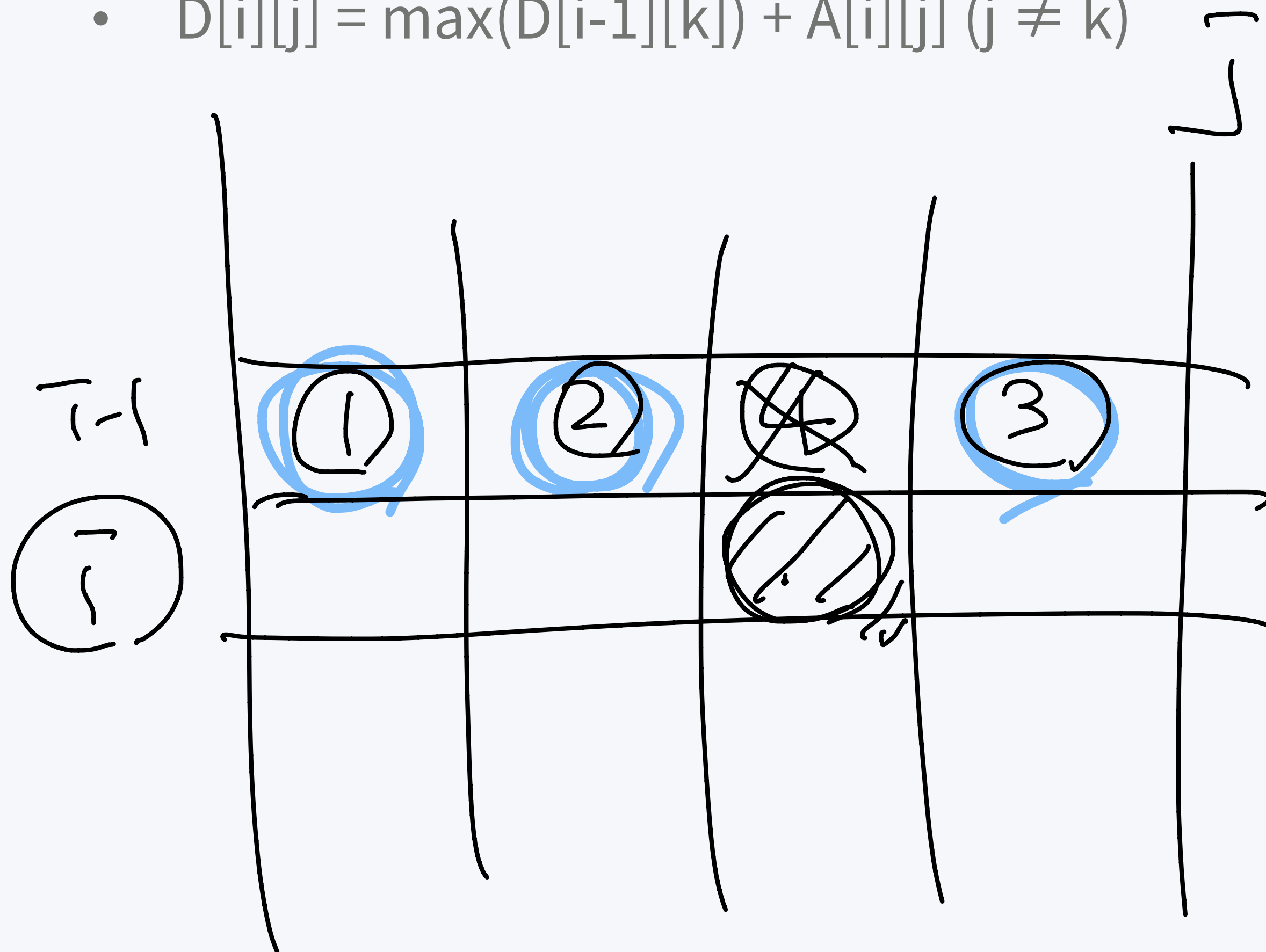


# 문제 5

<http://startl.in/2vPMxDC>

26

- $D[i][j] = (i, j)$ 에 도착했을 때, 얻을 수 있는 최대 점수
- $D[i][j] = \max(D[i-1][k]) + A[i][j] \ (j \neq k)$



$$D[i][j] = \max(D[i-1][k]) + A[i][j]$$

# 문제 5

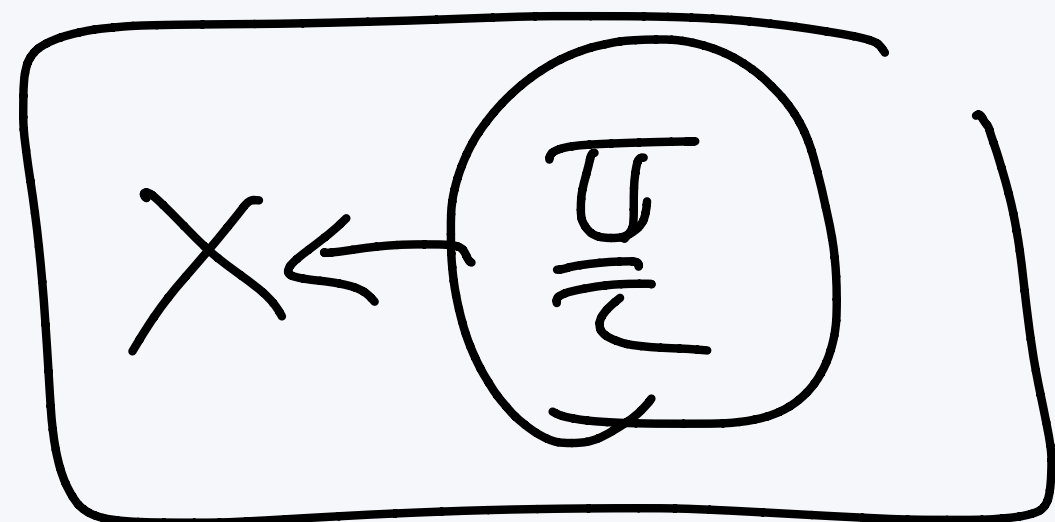
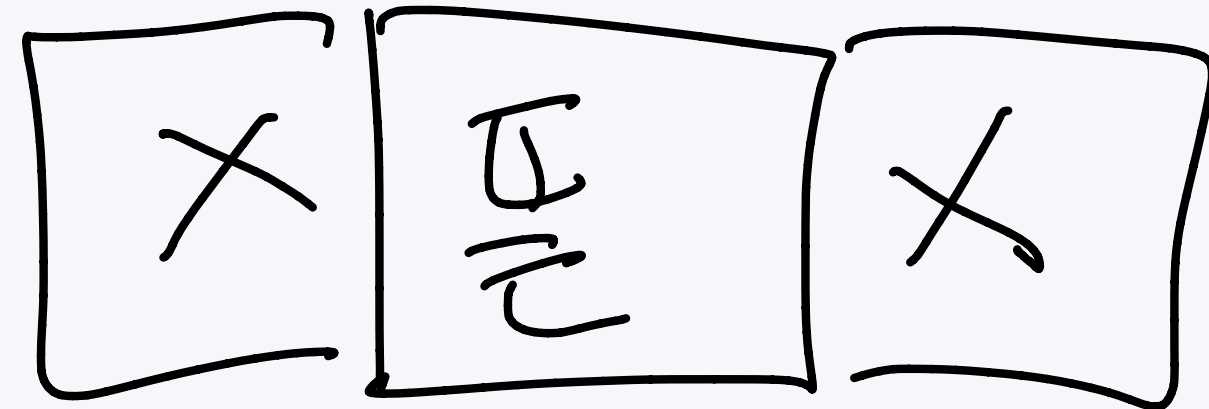
<http://startl.ink/2vPMxDC>

- C++: <https://gist.github.com/Baekjoon/dc8910c5b50aad8ae276fc2b59625c74>
- Java: <https://gist.github.com/Baekjoon/f72e0e61271529ed2581d0e0773d3716>

# 문제 6

<http://startl.in/2vPkLaD>

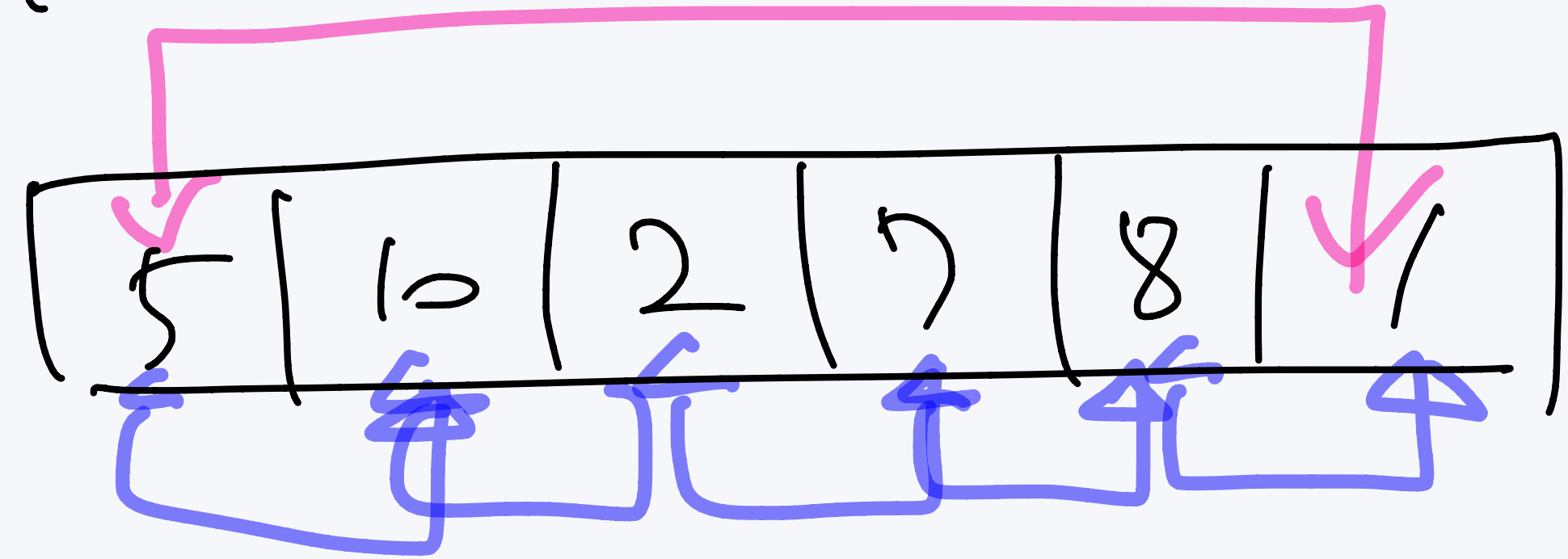
- N개의 스티커가 원형으로 연결되어 있다
- 각 스티커에는 점수가 있으며 A
- 뜯은 스티커 점수의 합을 최대로 하려고 한다
- $N \leq 100,000$



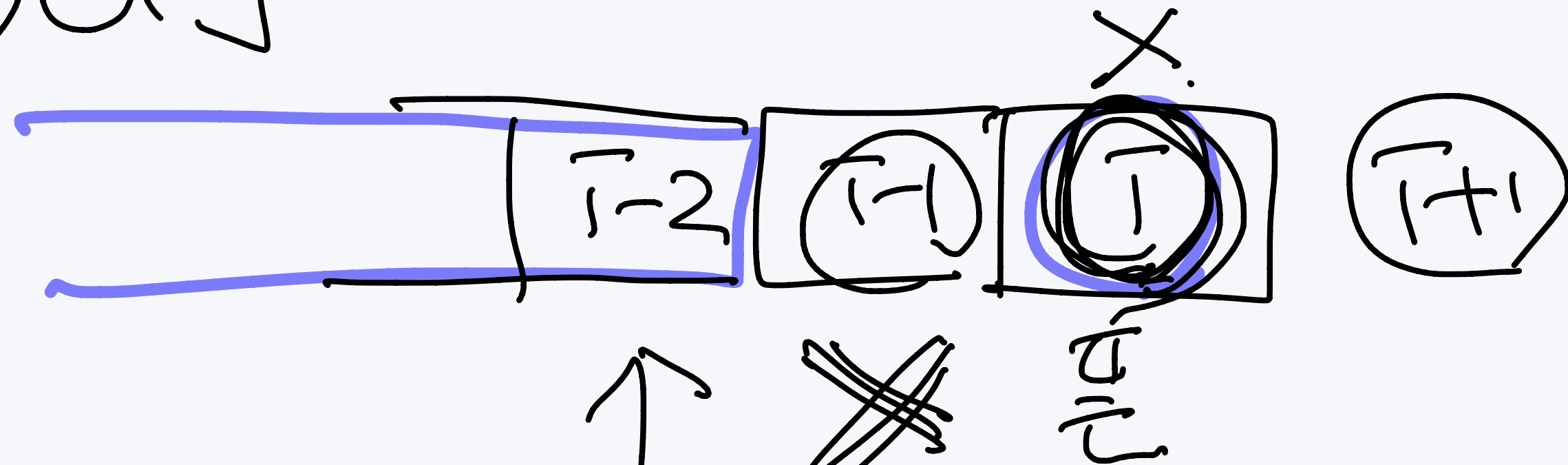
01 직선

N개 직선

차이



$DP[i] = 1 \sim i$ 번 스티커 점수의 최대



$(DP[i-2] + A[i], DP[i-1])$

# 문제 6

<http://startl.ink/2vPkLaD>

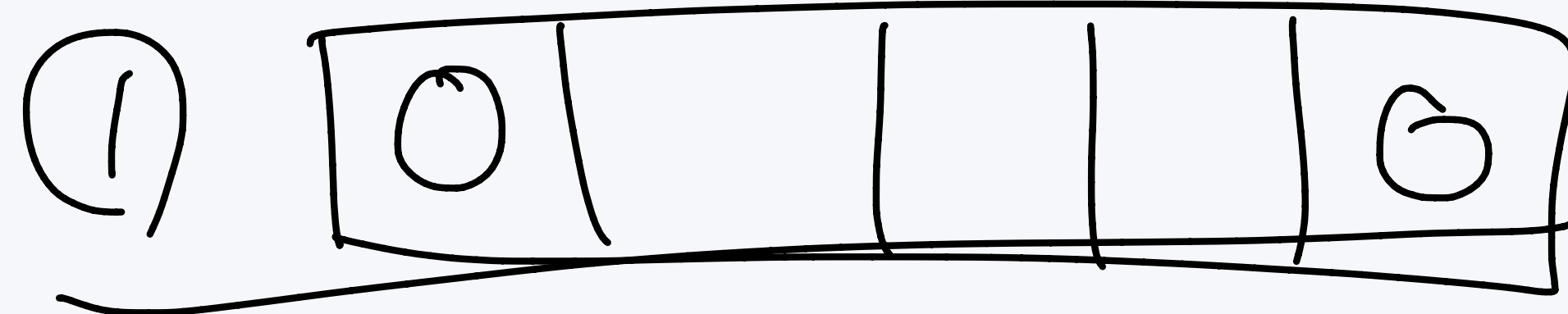
예외

$$n = 1$$

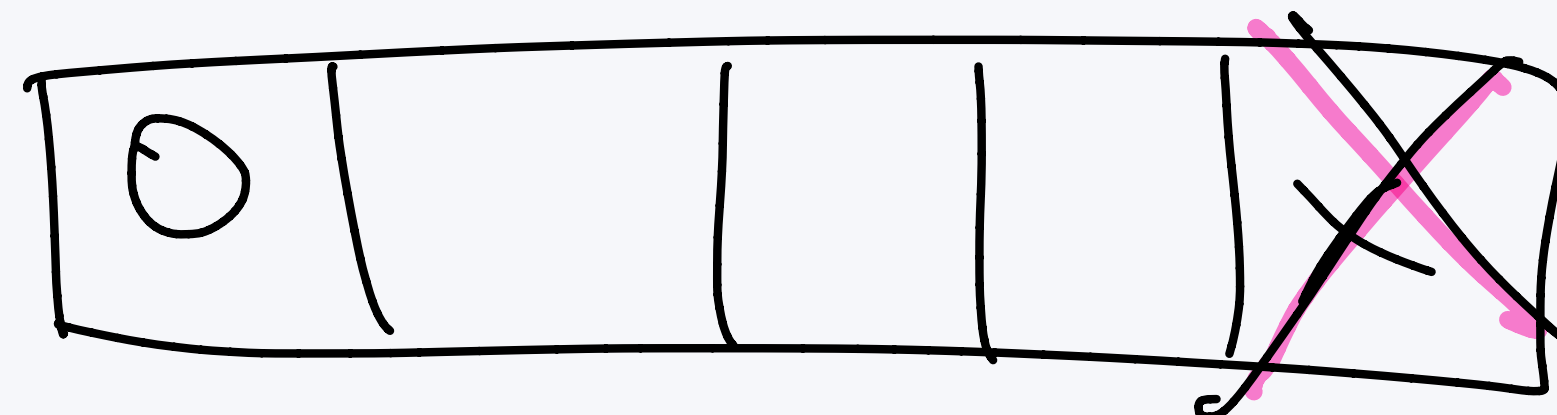
29

첫 = 마지막

- $D[i] = i$ 번 스티커까지 고려했을 때, 얻을 수 있는 최대 점수

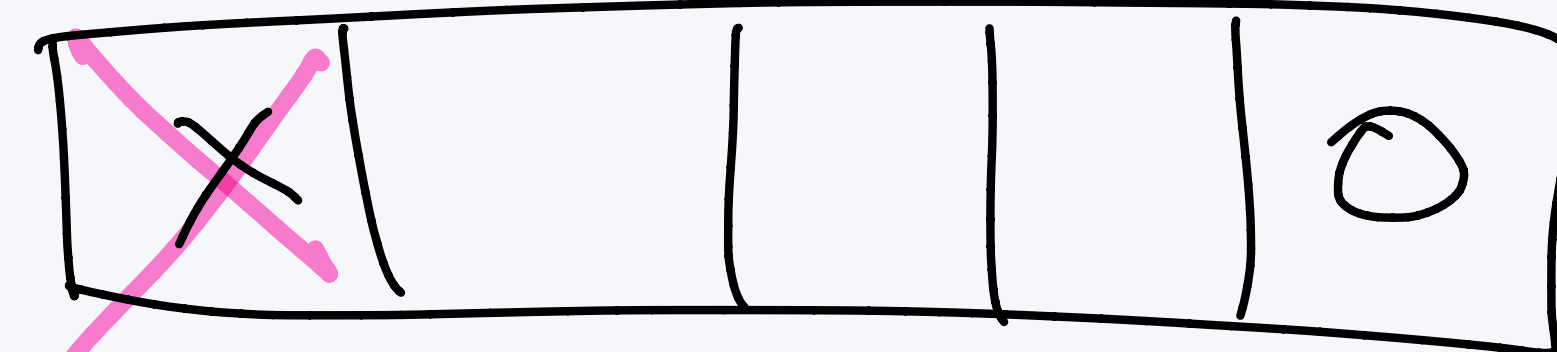


→ ②



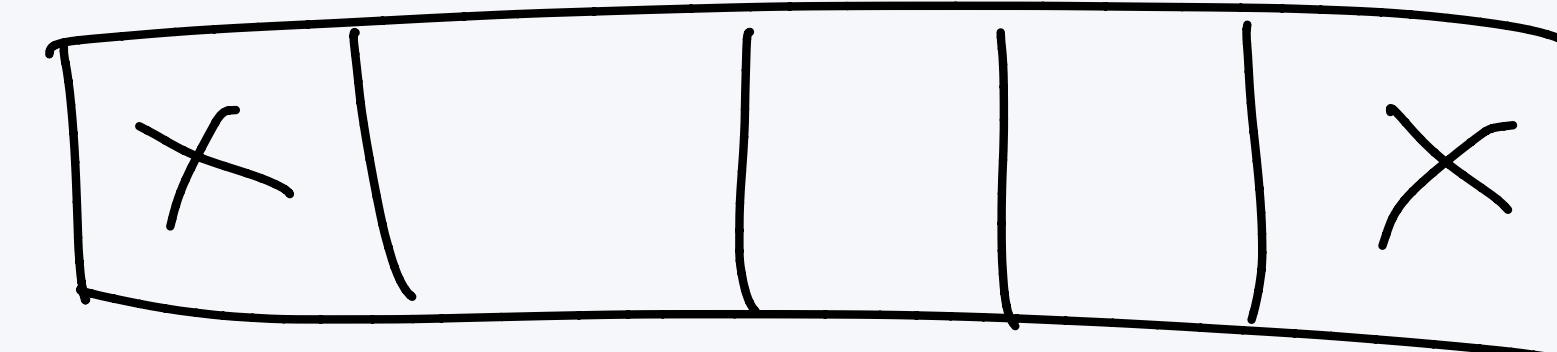
마지막 스티커 X

→ ③



첫 번째 스티커 X

→ ④



②, ③ 지니

# 문제 6

<http://startl.in/2vPkLaD>

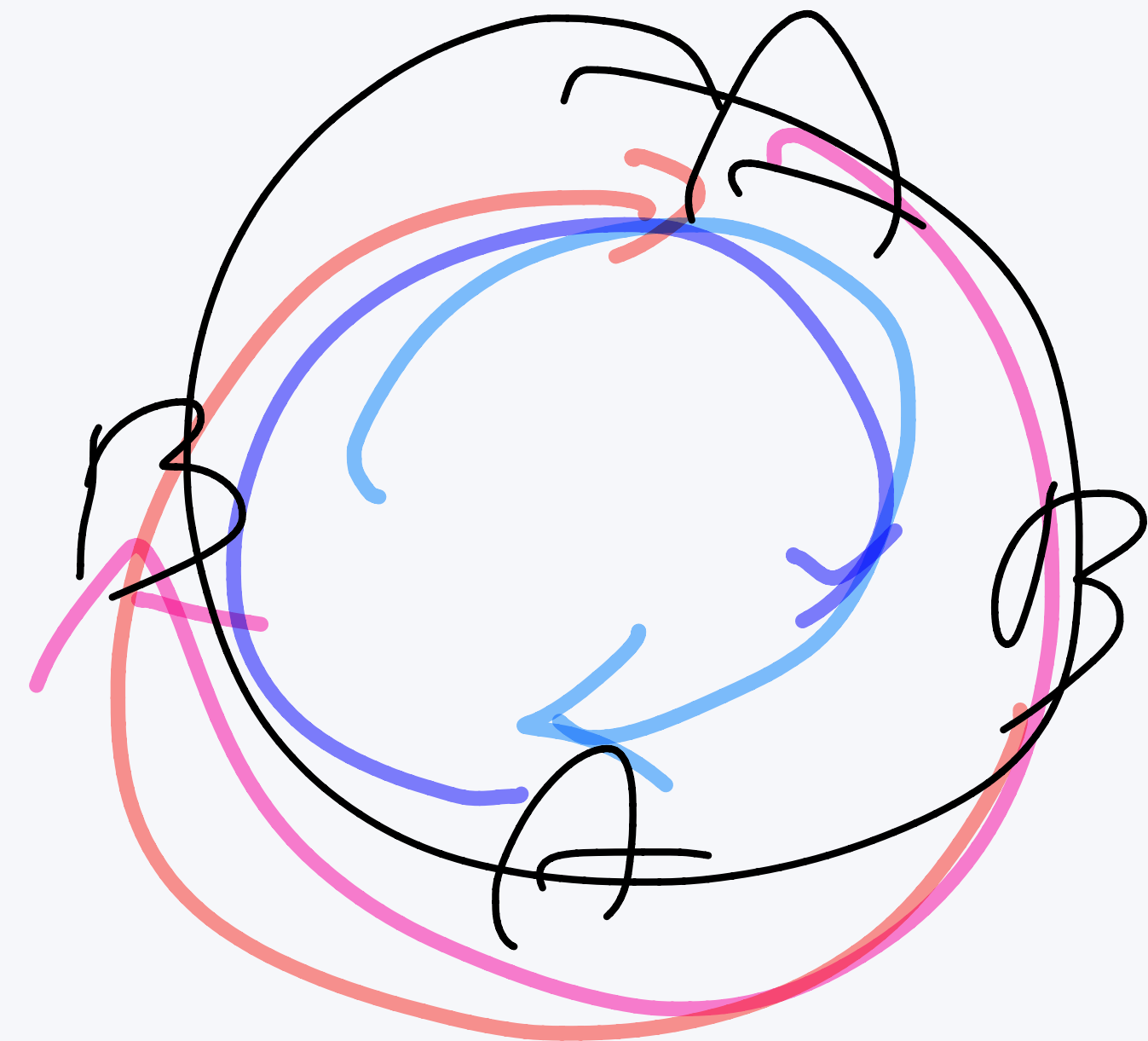
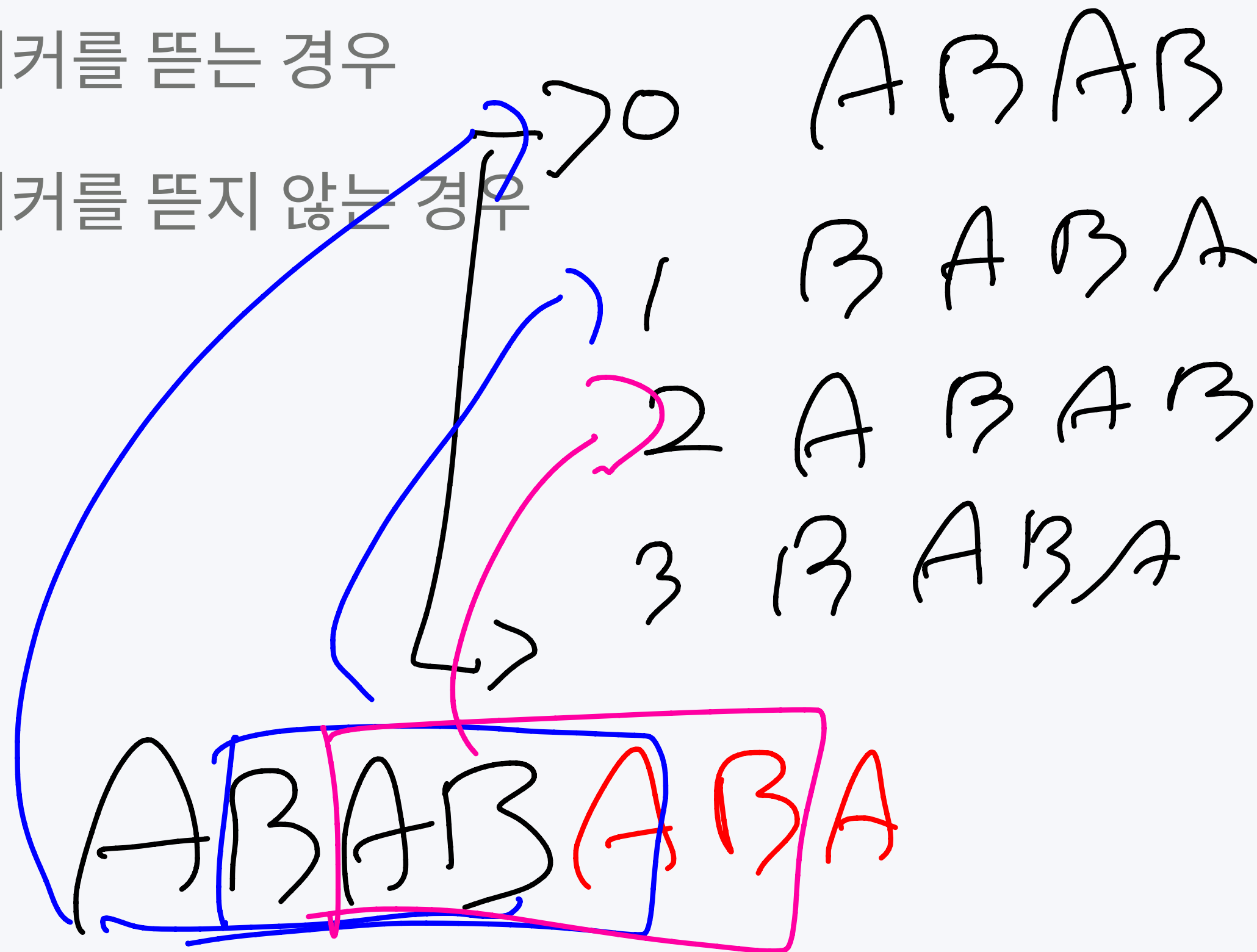
(N)

$S = \text{ABAB}$   
 $S \approx S \leftarrow$

$T = \text{BABABA}$

30

- $D[i] = i$ 번 스티커까지 고려했을 때, 얻을 수 있는 최대 점수
- $i$ 번 스티커를 뜯는 경우
- $i$ 번 스티커를 뜯지 않는 경우



# 문제 6

<http://startl.ink/2vPkLaD>

- $D[i]$  =  $i$ 번 스티커까지 고려했을 때, 얻을 수 있는 최대 점수
- $i$ 번 스티커를 뜯는 경우:  $D[i-2] + A[i]$
- $i$ 번 스티커를 뜯지 않는 경우:  $D[i-1]$

# 문제 6

<http://startl.ink/2vPkLaD>

- $D[i]$  =  $i$ 번 스티커까지 고려했을 때, 얻을 수 있는 최대 점수
- $i$ 번 스티커를 뜯는 경우:  $D[i-2] + A[i]$
- $i$ 번 스티커를 뜯지 않는 경우:  $D[i-1]$
- $D[i] = \max(D[i-1], D[i-2] + A[i])$



# 문제 6

<http://startl.ink/2vPkLaD>

- 그런데, 첫 스티커와 마지막 스티커가 붙어있으면 안된다

# 문제 6

<http://startl.ink/2vPkLaD>

- 그런데, 첫 스티커와 마지막 스티커가 붙어있으면 안된다
- 이 경우를 제외하기 위해서 첫 스티커를 제외하고 문제를 풀고, 마지막 스티커를 제외하고 문제를 푼다
- 예외는?

# 문제 6

<http://startl.ink/2vPkLaD>

- 그런데, 첫 스티커와 마지막 스티커가 붙어있으면 안된다
- 이 경우를 제외하기 위해서 첫 스티커를 제외하고 문제를 풀고, 마지막 스티커를 제외하고 문제를 푼다
- 예외는?  $n = 1$ 인 경우

# 문제 6

<http://startl.ink/2vPkLaD>

- C++: <https://gist.github.com/Baekjoon/a7d710f03457a6a1162fb5d67a0f1cbe>
- Java: <https://gist.github.com/Baekjoon/9da49e7379f76084aec2a6a922f85ff6>

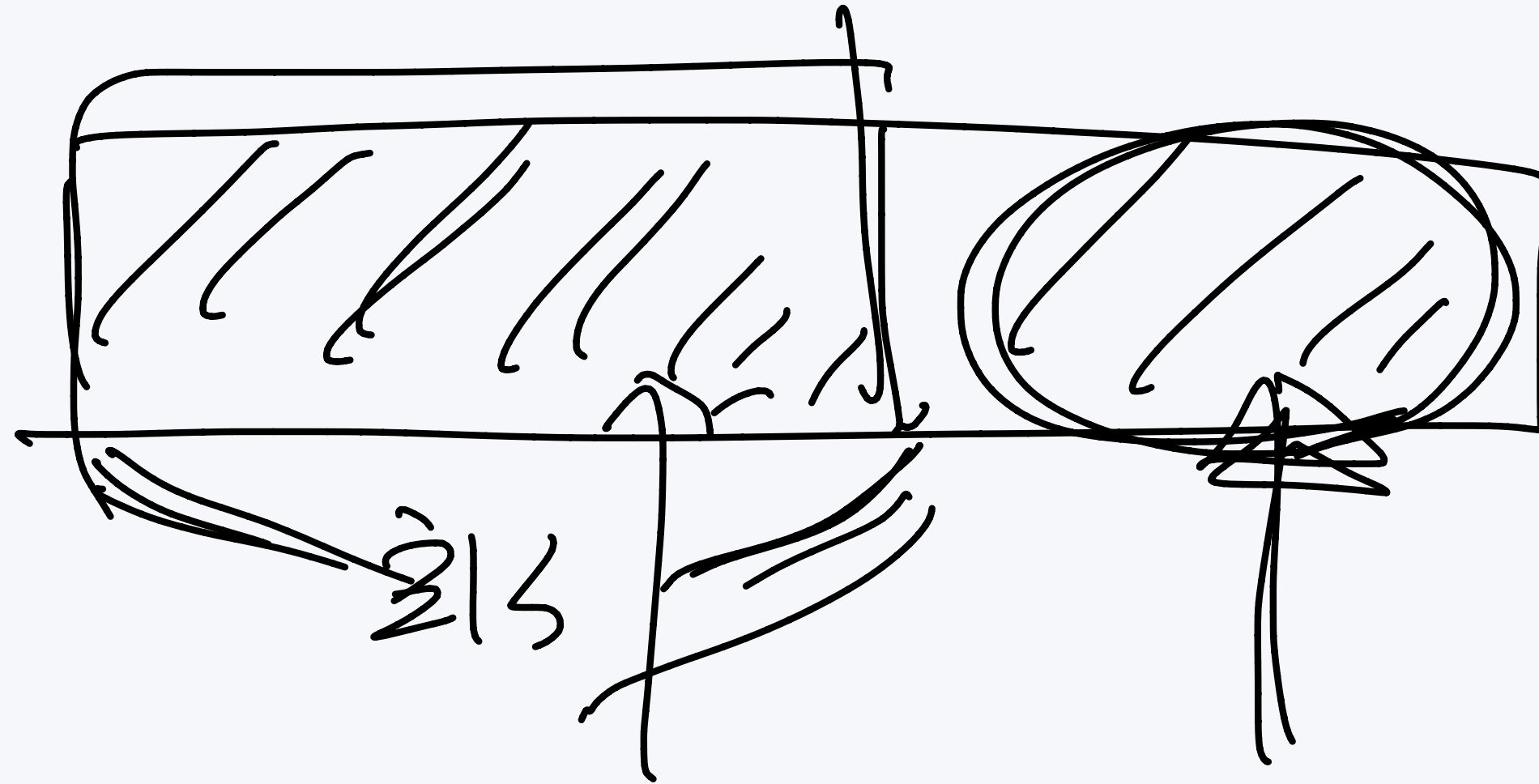
# 문제 7

<http://startl.ink/2j7JWPA>

37



- 단어 S가 있고, 단어 조각 A가 있다
- 단어 조각을 이용해서 단어 S를 만들 때 필요한 최소 단어 개수를 구하는 문제



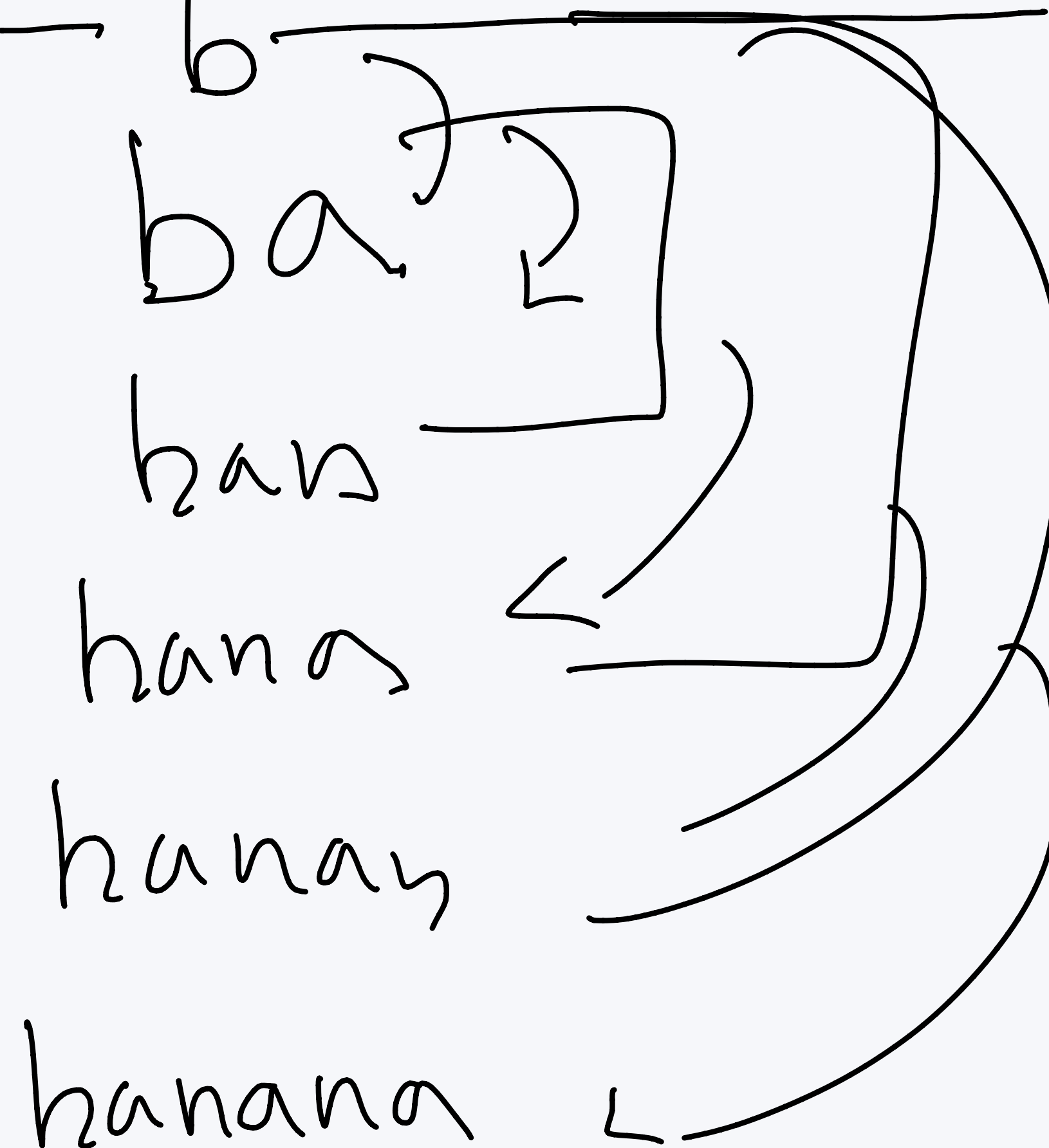
# 문제 7

<http://startl.in/2j7JWPA>

banana

38


- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수



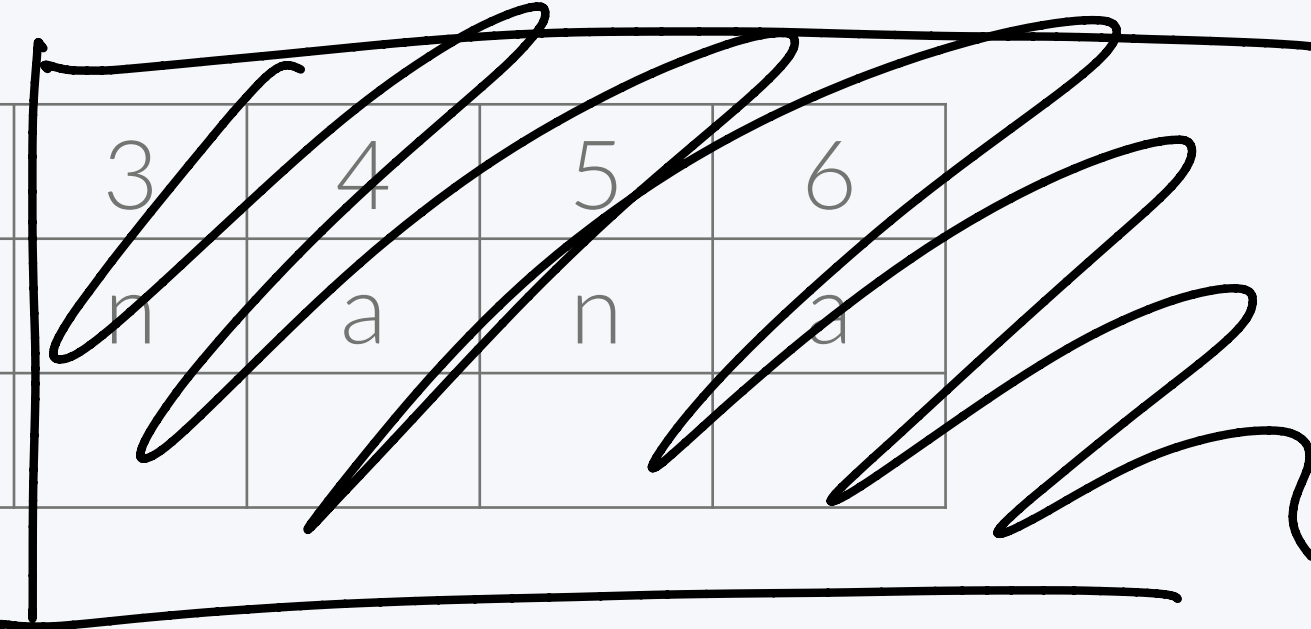
# 문제 7

<http://startl.in/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"



i	0	1	2	3	4	5	6
s[i]		b	a	n	a	n	a
d[i]	0						



# 문제 7

<http://startl.ink/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"

i	0	1	2	3	4	5	6
s[i]		b	a	n	a	n	a
d[i]	0	-1					
a[0]	b	a					
a[1]	n	a					
a[2]		n					
a[3]		a					



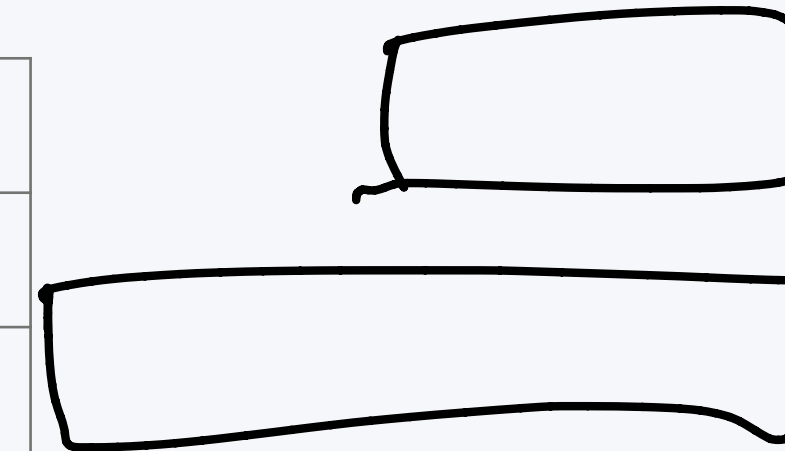
# 문제 7

41

<http://startl.ink/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"

i	0	1	2	3	4	5	6
s[i]		<del>b</del>	a	n	a	n	a
d[i]	0	-1	1				
a[0]		b	a				
a[1]		n	a				
a[2]			n				
a[3]			a				



# 문제 7

<http://startl.in/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"

i	0	1	2	3	4	5	6
s[i]		b	a	<del>n</del>	a	n	a
d[i]	0	-1	1	2			
a[0]			b	a	174 ba n		
a[1]			n	a	1 (2r)		
a[2]				n	+1		
a[3]				a	1		

# 문제 7

43

<http://startl.in/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"

i	0	1	2	3	4	5	6
s[i]		b	a	n	a	n	a
d[i]	0	-1	1	2	2		
a[0]				b	a		
a[1]				n	a		
a[2]					n		
a[3]					a		

Handwritten annotations on the table:

- A horizontal box highlights the characters 'b', 'a', 'n' in the s[i] row for i=1, 2, 3.
- A horizontal box highlights the characters 'n', 'a' in the a[1] row for i=3, 4.
- A horizontal box highlights the character 'a' in the a[3] row for i=4.
- Arrows point from the circled '1' in d[i] at i=2 to the 'b' in a[0] at i=3, and from the circled '2' in d[i] at i=3 to the 'n' in a[1] at i=3.
- An arrow points from the circled '2' in d[i] at i=4 to the 'a' in a[3] at i=4.
- A curved arrow points from the circled 'a' in a[3] at i=4 to the 'a' in a[0] at i=4.
- The number '3' is written next to the 'a' in a[0] at i=4.
- The text '+1' is written next to the 'a' in a[1] at i=4.
- The text '+1' is written next to the 'a' in a[3] at i=4.

# 문제 7

<http://startl.in/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"

i	0	1	2	3	4	5	6
s[i]		b	a	n	a	n	a
d[i]	0	-1	1	2	2	3	
a[0]					b	a	
a[1]					n	a	
a[2]						n	
a[3]						a	

# 문제 7

<http://startl.ink/2j7JWPA>

- $D[i]$  = S의 단어 i번째 문자열까지 만드는 필요한 단어 조각의 최소 개수
- 단어: "ba", "na", "n", "a"

길이 20

i	0	1	2	3	4	5	6
s[i]		b	a	n	a	n	a
d[i]	0	-1	1	2	2	3	3
a[0]						b	a
a[1]						n	a
a[2]							n
a[3]							a

↓

# 문제 7

<http://startl.ink/2j7JWPA>

- C++: <https://gist.github.com/Baekjoon/b4033687ee6f1812343b6493fc5bf4b9>
- Java: <https://gist.github.com/Baekjoon/eabb0b67e24d6b4b6021fd9a21d8f803>

# 스티커

$D[i][j] = 2 \times \text{스티커}$   
 T열의 상태 j이다

X	X	0
X	0	X
0	1	2
상대		

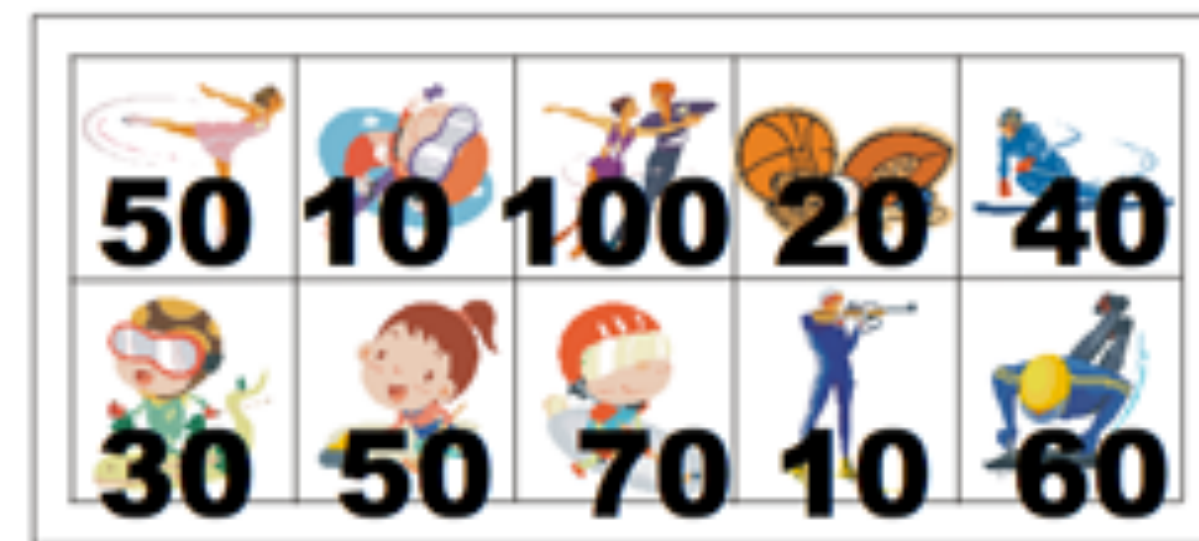
<https://www.acmicpc.net/problem/9465>

- 스티커 2n개가  $2 \times n$  모양으로 배치되어 있다
- 스티커 한 장을 떼면 변을 공유하는 스티커는 모두 찢어져서 사용할 수 없다
- 점수의 합을 최대로 만드는 문제

$$D[i][0] = \max(D[i-1][0], D[i-1][1], D[i-1][2])$$



(a)



(b)

$$D[i][1] = \max(D[i-1][0], D[i-1][2]) + A[i][1]$$

0	X	0	X
X	0	1	X
1	0	2	0

# 스티커

<https://www.acmicpc.net/problem/9465>

- $D[i][j] = 2 \times i$  에서 얻을 수 있는 최대 점수,  $i$  번 열에서 뜯는 스티커는  $j$
- $j = 0 \rightarrow$  뜯지 않음
- $j = 1 \rightarrow$  위쪽 스티커를 뜯음
- $j = 2 \rightarrow$  아래쪽 스티커를 뜯음



# 스티커

<https://www.acmicpc.net/problem/9465>

- $D[i][j] = 2 \times i$  에서 얻을 수 있는 최대 점수,  $i$  번 열에서 뜯는 스티커는  $j$
- 뜯지 않음 ( $D[i][0]$ )
  - $i-1$  열에서 스티커를 어떻게 뜯었는지 상관이 없다
  - $\max(D[i-1][0], D[i-1][1], D[i-1][2])$
- 위쪽 스티커를 뜯음 ( $D[i][1]$ )
  - $i-1$  열에서 위쪽 스티커는 뜯으면 안된다
  - $\max(D[i-1][0], D[i-1][2]) + A[i][0]$
- 아래쪽 스티커를 뜯음 ( $D[i][2]$ )
  - $i-1$  열에서 아래쪽 스티커는 뜯으면 안된다
  - $\max(D[i-1][0], D[i-1][1]) + A[i][1]$

# 스티커

50

<https://www.acmicpc.net/problem/9465>

- C/C++: <https://gist.github.com/Baekjoon/3fec4c0ef968b943041a>
- Java: <https://gist.github.com/Baekjoon/a05b5a7ef6e5cd8f7951>

# 암호 코드

<https://www.acmicpc.net/problem/2011>

- 어떤 암호가 주어졌을 때, 그 암호의 해석이 몇 가지가 나올 수 있는지 구하는 문제

1 ~ 26

- A는 1, B는 2, ..., Z는 26
- BEAN → 25114  
B E A N
- 25114 → "BEAAD", "YAAD", "YAN", "YKD", "BEKD", "BEAN"  
25 1 1 4    25 1 1 4    25 1 1 4  
251 14

# 암호 코드

<https://www.acmicpc.net/problem/2011>

- $D[i]$  =  $i$ 번째 문자까지 해석했을 때, 나올 수 있는 해석의 가짓수
- $i$ 번째 문자에게 가능한 경우
- 1자리 암호
- 2자리 암호

# 암호 코드

53

<https://www.acmicpc.net/problem/2011>

- $D[i]$  =  $i$ 번째 문자까지 해석했을 때, 나올 수 있는 해석의 개짓수
- $i$ 번째 문자에게 가능한 경우
- 1자리 암호
  - 0을 제외
- 2자리 암호
  - $10 \leq x \leq 26$

이항

$$\frac{(A/B)^2 \% C}{(6/3)^2 \% 3 = 2}$$
$$(0/0)^2 \% 3$$
$$(A \times B^{C-2}) \% C$$

# 암호 코드

<https://www.acmicpc.net/problem/2011>

$\% M$

`d[0] = 1;`

```
for (int i=1; i<=n; i++) {
    int x = s[i] - '0';
    if (1 <= x && x <= 9) {
        d[i] = (d[i] + d[i-1]) % mod;
    }
}
```

`if (i==1) continue;`

`if (s[i-1] == '0') continue;`

`x = (s[i-1] - '0') * 10 + (s[i] - '0');`

`if (10 <= x && x <= 26) {`  
`d[i] = (d[i] + d[i-2]) % mod;`  
`}`

}

A-I

(I)

~9

$$(A+B) \% C$$

$$= (A \% C + B \% C) \% C$$

$$(A-B) \% C$$

$$(A \% C - B \% C + C) \% C$$

$$(6-5) \% 3 = 1$$

$$(0-2) \% 3$$

$$= -2 \% 3 =$$

$$= 1$$

$$(A+B) \% C =$$

# 암호 코드

<https://www.acmicpc.net/problem/2011>

- C: <https://gist.github.com/Baekjoon/33126f8bcbfa55ebc9b1>
- C++: <https://gist.github.com/Baekjoon/dc071a89a81cb88fb887>
- Java: <https://gist.github.com/Baekjoon/725d19f289d7dda4a191>

# 맹세

56

<https://www.acmicpc.net/problem/3407>

최소

- 문장이 주어졌을 때, 원소기호의 조합으로만 나타낼 수 있는지 아닌지 구하는 문제

H He Li



# 맹세

<https://www.acmicpc.net/problem/3407>

- $D[i]$  = 문장의  $i$ 번째 문자까지를 원소 기호의 조합으로 나타낼 수 있으면 1, 아니면 0

# 맹세

<https://www.acmicpc.net/problem/3407>

- $D[i]$  = 문장의  $i$ 번째 문자까지를 원소 기호의 조합으로 나타낼 수 있으면 1, 아니면 0
- $D[i-1]$ 이 true이고,  $A[i]$ 가 원소 기호이면  $D[i] = \text{true}$
- $D[i-2]$ 가 true이고,  $A[i-1], A[i]$ 가 원소 기호이면  $D[i] = \text{true}$

# 맹세

<https://www.acmicpc.net/problem/3407>

- C++: <https://gist.github.com/Baekjoon/c4c825c7d05b1dbd27665e946a9b36b2>
- Java: <https://gist.github.com/Baekjoon/4a283f3da7122f761fdeef538e529c7e>

# BOJ 거리

<https://www.acmicpc.net/problem/12026>

- 스타트는 BOJ를 외치면서 링크를 만나러 가려고 한다
- 따라서, 스타트는 B, O, J, B, O, J, B, O, J, ... 순서로 보도블럭을 밟으면서 점프를 할 것이다
- 스타트가 링크를 만나는데 필요한 에너지 양의 최소값을 구하는 프로그램을 작성하시오.

# BOJ 거리

<https://www.acmicpc.net/problem/12026>

- $D[N]$  = N에 오는 최소 에너지

# BOJ 거리

62

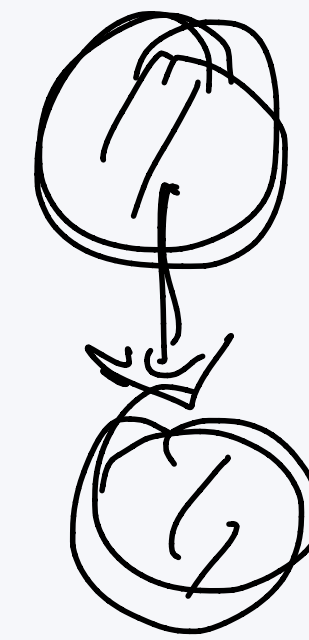
<https://www.acmicpc.net/problem/12026>

- C++: <https://gist.github.com/Baekjoon/d5648836057d6621aaa8baacb5762160>
- Java: <https://gist.github.com/Baekjoon/b5e6050d7ca94c08f3f6ddde384a6665>

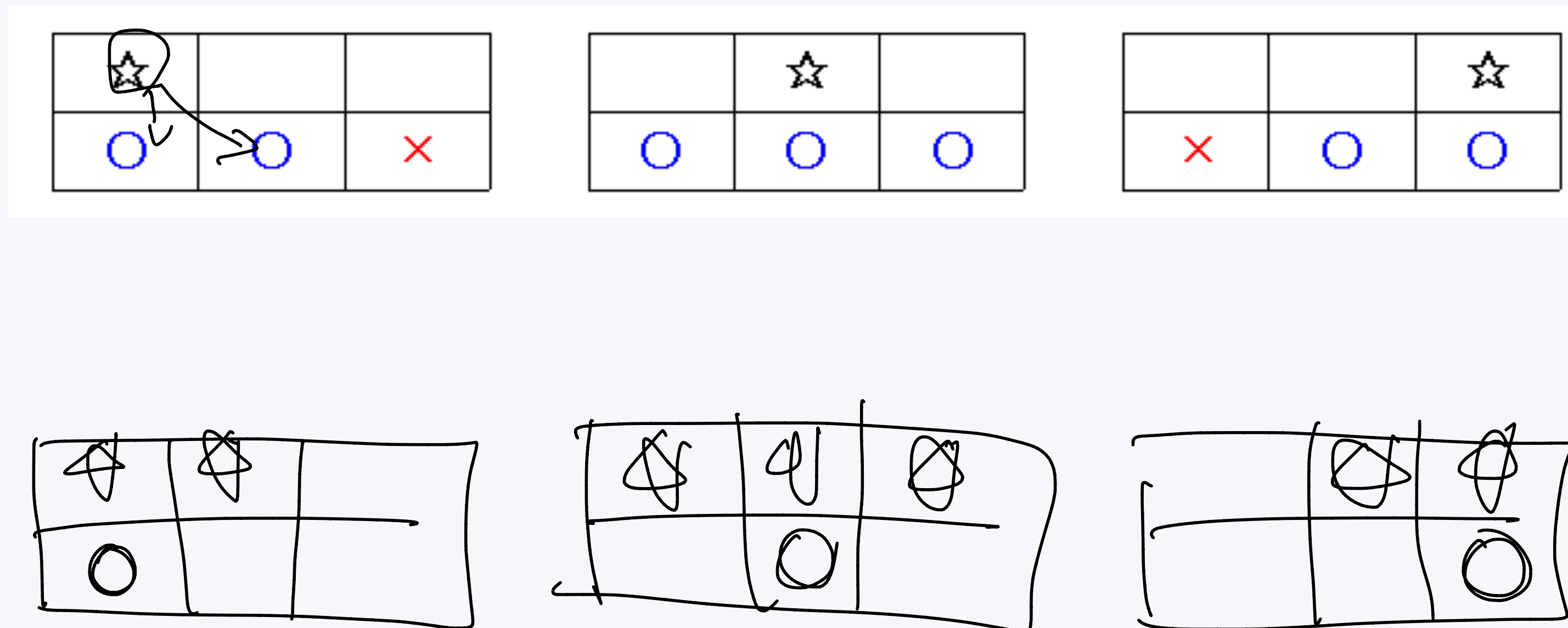
# 내려가기

<https://www.acmicpc.net/problem/2096>

63



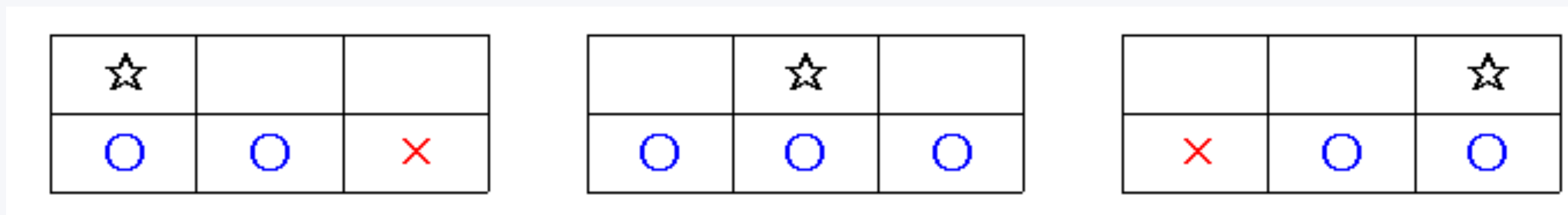
- N줄에 0이상 9이하의 숫자가 세 개씩 적혀져 있다
- 첫 줄에서 시작해서 마지막 줄에서 끝난다
- 처음에 세 개의 숫자 중에 하나를 골라서 시작하고, 다음 줄로 내려갈 땐 붙어있는 수로만 이동 가능



# 내려가기

<https://www.acmicpc.net/problem/2096>

- $D[i][j] = (i, j)$ 에 도착할 때, 얻을 수 있는 최대 점수
- $D[i][0] = \max(D[i-1][0], D[i-1][1]) + A[i][0]$
- $D[i][1] = \max(D[i-1][0], D[i-1][1], D[i-1][2]) + A[i][1]$
- $D[i][2] = \max(D[i-1][1], D[i-1][2]) + A[i][2]$





# 내려가기

<https://www.acmicpc.net/problem/2096>

- 메모리 제한이 4MB이기 때문에, 공간을 조금만 사용해야 한다

# 내려가기

<https://www.acmicpc.net/problem/2096>

- C++: <https://gist.github.com/Baekjoon/d1e50b6ee5f618d9660bcce51348335d>
- Java: <https://gist.github.com/Baekjoon/ec7376aa87c0834cc14095f13ec1f6b9>

# 단어 섞기

<https://www.acmicpc.net/problem/9177>

67

A, B, C  
순서 섞기

- 세 개의 단어가 주어졌을 때
- 첫 번째와 두 번째 단어를 섞어서 세 번째 단어를 만들 수 있는지 없는지 구하는 문제
- 이 때, 원래 단어에 있는 순서가 바뀌면 안된다 574
- cat, tree의 경우
- tcraete와 catrtee는 만들 수 있지만, cttaree를 만들 수는 없다
- 단어의 길이는 200을 넘지 않는다

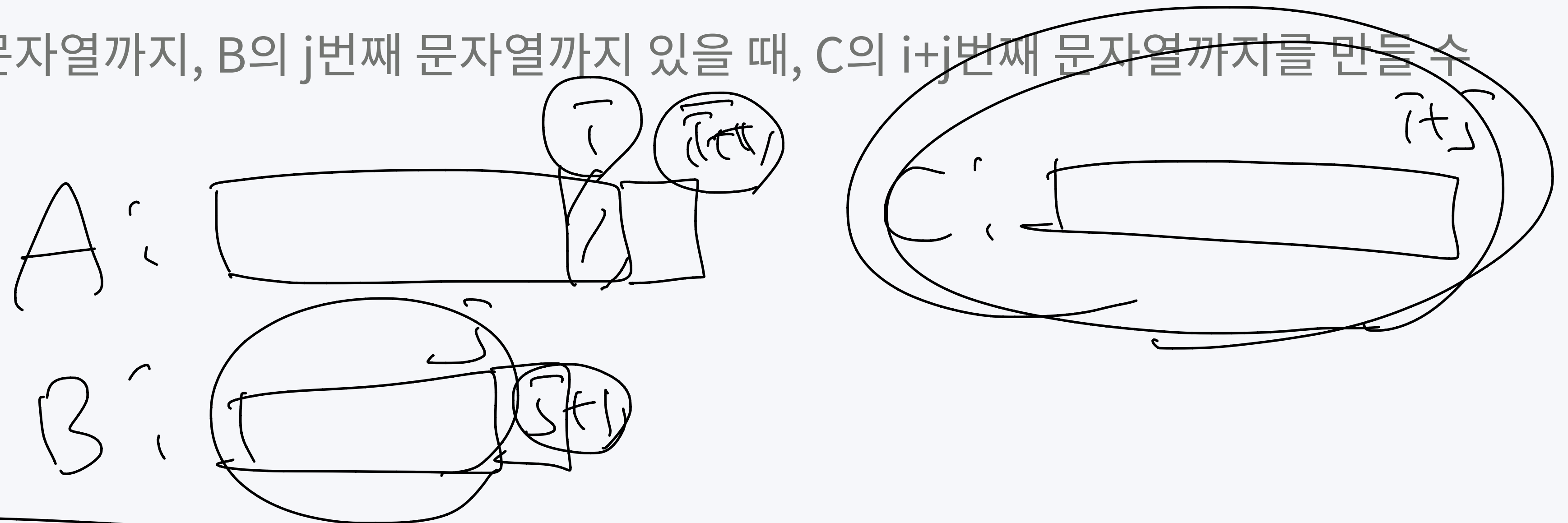
A = cat  
B = tree  
A = at  
B = tree  
atrtee

A = cat  
B = tree  
tcraete  
catrtee  
A = cat  
B = tree  
catrtee

# 단어 섞기

<https://www.acmicpc.net/problem/9177>

- $D[i][j]$  = A의 i번째 문자열까지, B의 j번째 문자열까지 있을 때, C의 i+j번째 문자열까지를 만들 수 있으면 1, 없으면 0



경우의 수 : 27121

$A \frac{2}{2} \frac{14}{2}$

$B \frac{2}{3} \frac{6}{3}$

# 단어 섞기

<https://www.acmicpc.net/problem/9177>

- $A[i]$ 와  $C[i+j]$ 가 같으면, 이제  $D[i+1][j]$ 의 값을 살펴봐야 한다
- $B[j]$ 와  $C[i+j]$ 가 같으면, 이제  $D[i][j+1]$ 의 값을 살펴봐야 한다

# 단어 섞기

<https://www.acmicpc.net/problem/9177>

- C++: <https://gist.github.com/Baekjoon/c679d94ee649354cbef985358fde8d1a>
- Java: <https://gist.github.com/Baekjoon/799359029f444d451f65f3c764f7ee9e>

# 단어 격자

<https://www.acmicpc.net/problem/2418>

K번째글자까지 단어를 맞추기  
(1, 3)

71

- 단어 격자와 단어가 주어졌을 때, 주어진 단어를 읽는 방법의 경우의 수를 구하는 문제
- 첫 글자는 격자의 어느 곳이어도 되고, 두 번째 글자부터는 그 전 글자가 있던 칸과 인접한 칸이어야 한다
- 인접: 상하좌우, 대각선 8칸

E	R	A	T
A	T	S	R
A	U	T	U

TIARTU

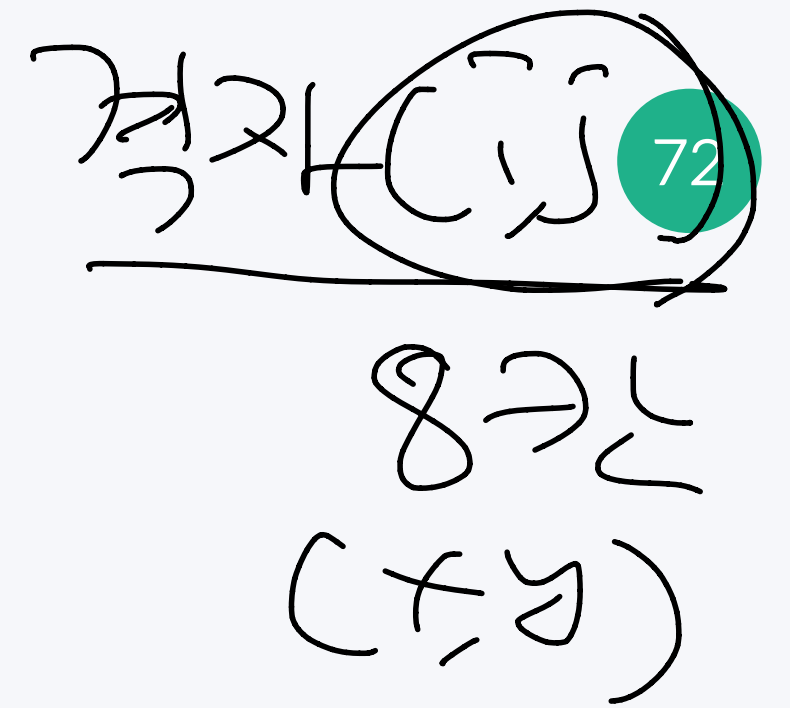
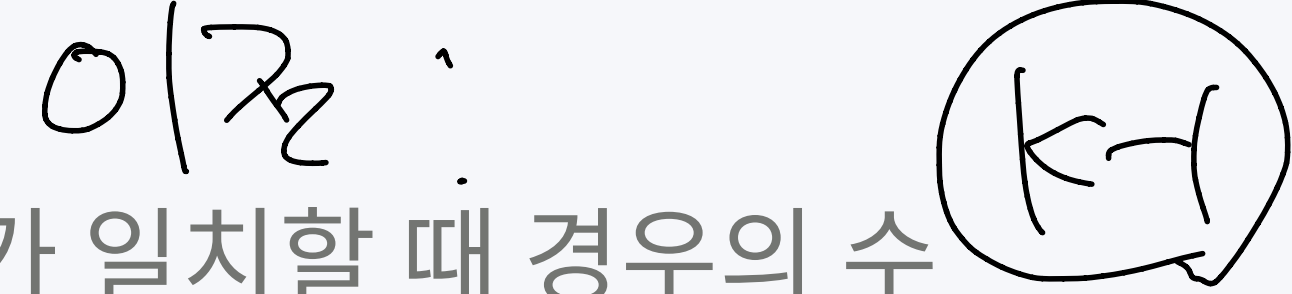
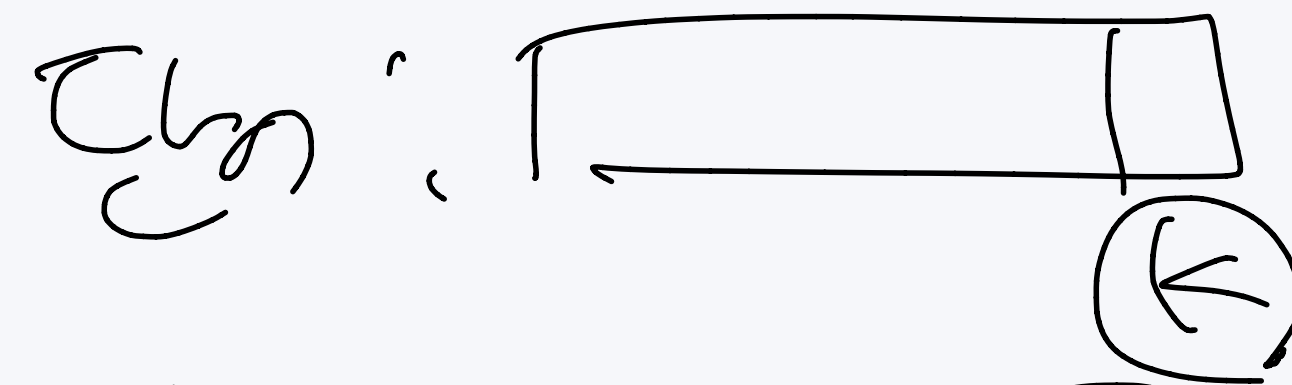
8방향

# 단어 격자

<https://www.acmicpc.net/problem/2418>

- $D[k][i][j]$  = 주어진 단어의 k번째 글자와 격자의 (i, j)가 일치할 때 경우의 수

$$D[k][i][j] = \sum D[k+1][x][y]$$





# 단어 격자

<https://www.acmicpc.net/problem/2418>

- $D[k][i][j]$  = 주어진 단어의  $k$ 번째 글자와 격자의  $(i, j)$ 가 일치할 때 경우의 수
- $D[k][i][j] = \sum D[k-1][x][y]$
- 조건
- $(i, j)$ 와  $(x, y)$ 가 인접
- 단어의  $k-1$ 번째 글자와 격자의  $(x, y)$ 에 있는 글자가 일치

# 단어 격자

<https://www.acmicpc.net/problem/2418>

- C++: <https://gist.github.com/Baekjoon/8f82fb76cbc8ccef5503cb03156b852a>
- Java: <https://gist.github.com/Baekjoon/8f0229870728b07415388afe4ba2ef86>

# 단어 게임

75

<https://www.acmicpc.net/problem/1856>

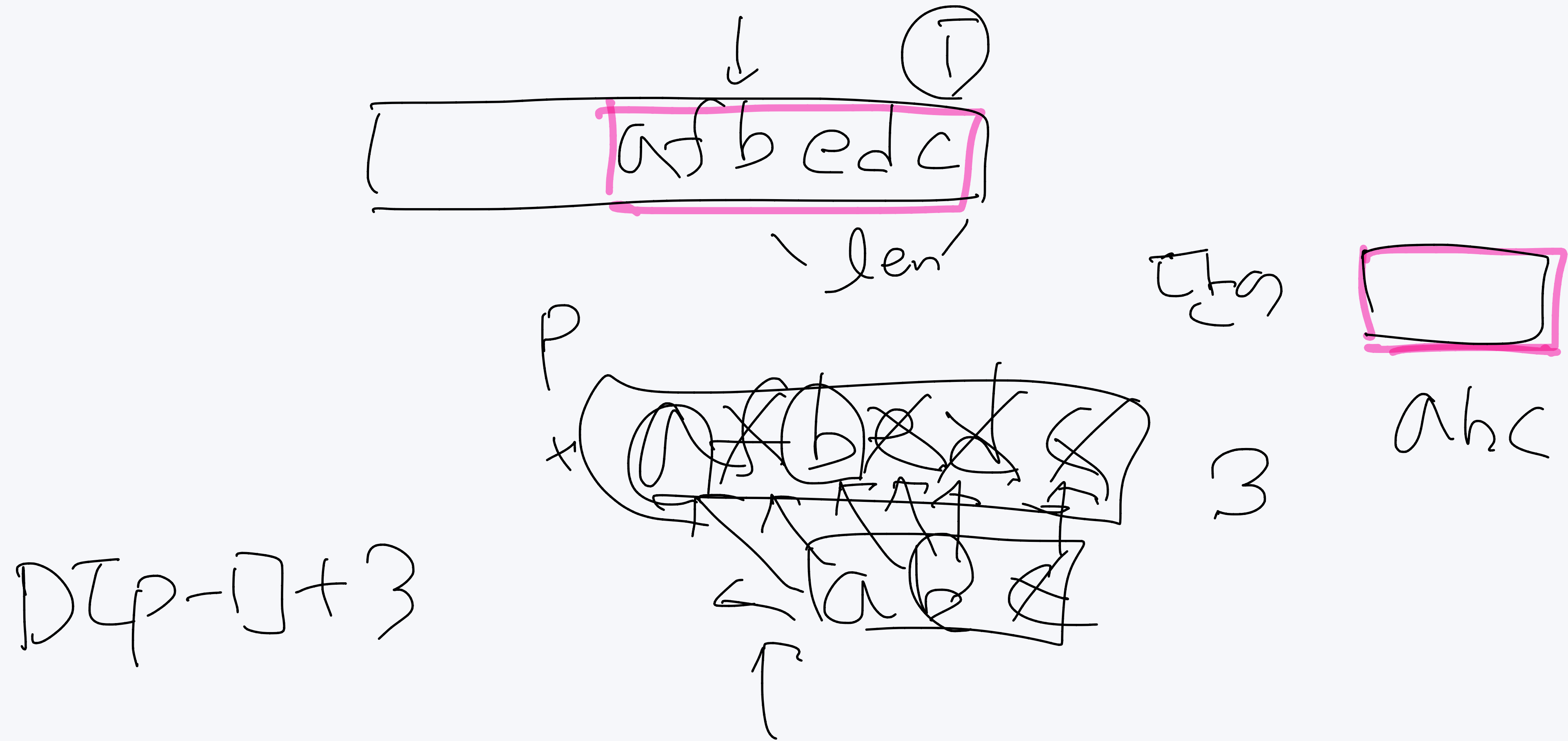
- 단어가  $w$ 개인 사전이 주어진다 (길이  $\leq 25$ )
- 문자열  $S$ 를 사전에 있는 단어의 조합으로 나타내려고 한다
- 이 때, 문자열에서 제거해야 하는 문자의 최소 개수

(

# 단어 게임

<https://www.acmicpc.net/problem/1856>

- $D[i]$  = 문자열 S의 i번째 문자열까지를 나타내는데 제거해야하는 문자의 최소 개수



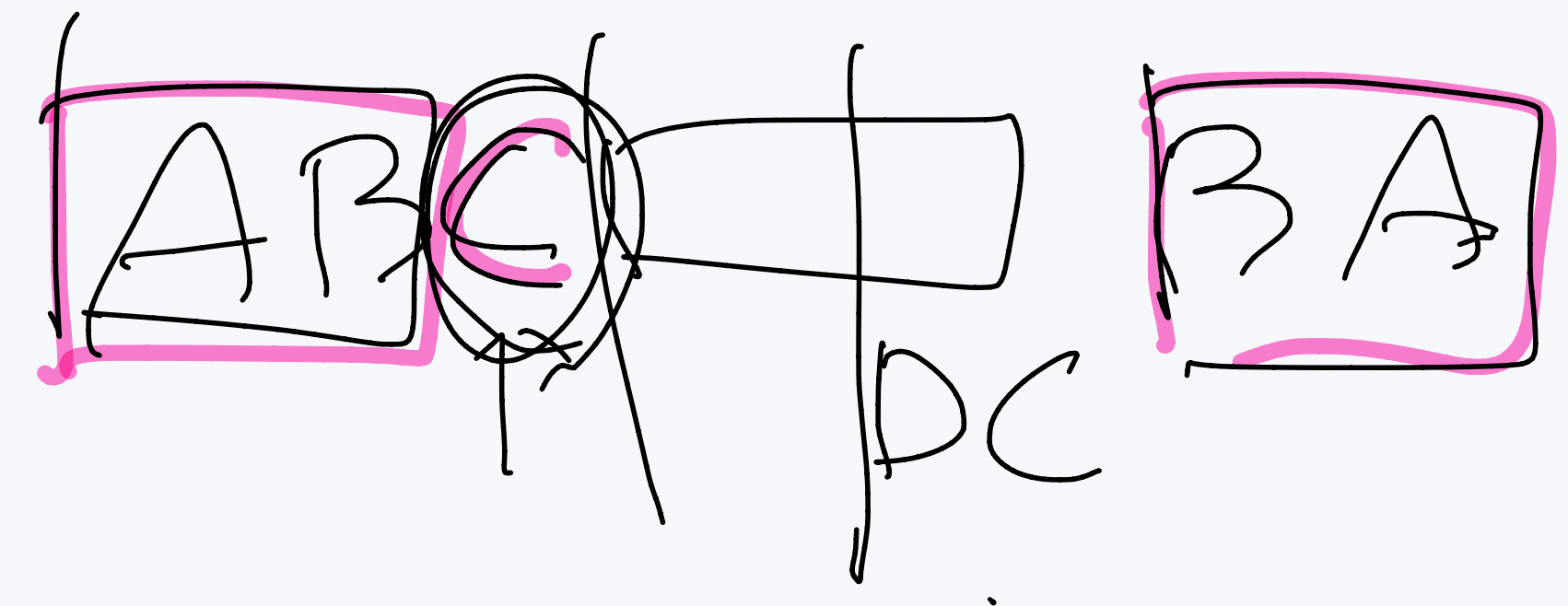
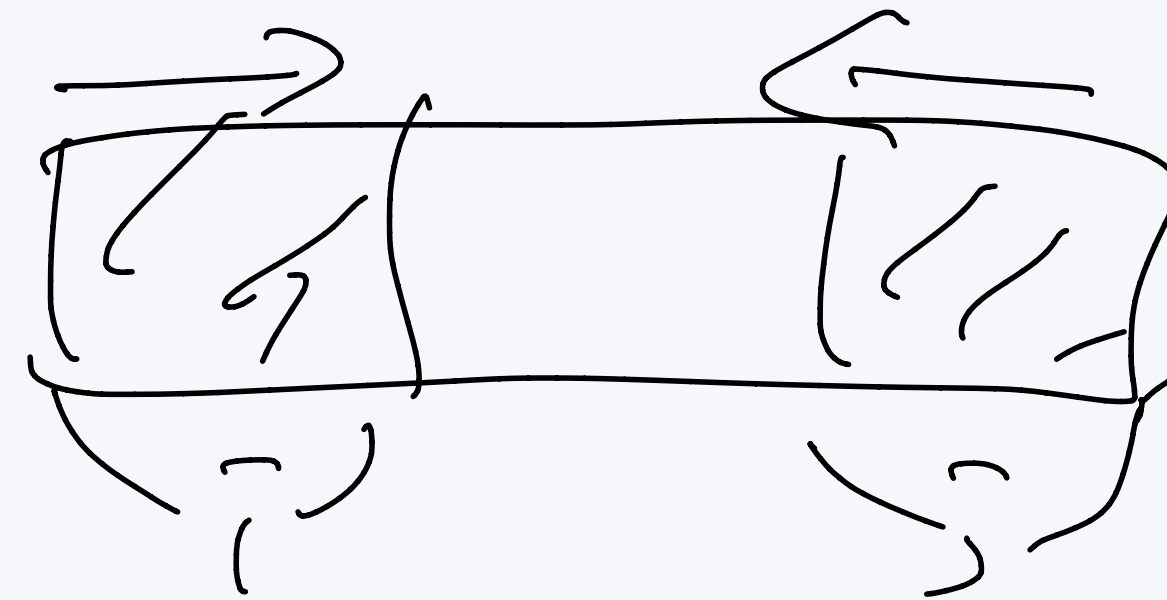
# 단어 게임

D[0][0] S[0]

77

<https://www.acmicpc.net/problem/1856>

- C++: <https://gist.github.com/Baekjoon/a867be8d57a0a717e7b0d21f3f90bee5>
- Java: <https://gist.github.com/Baekjoon/3c65b747f9b86a35e445a497ba51e29c>



D[ len ][ str ] [ dir ]  
2 1 0 |                      ↑                      str ← 0  
2 1 0 |                      ↓                      → 1