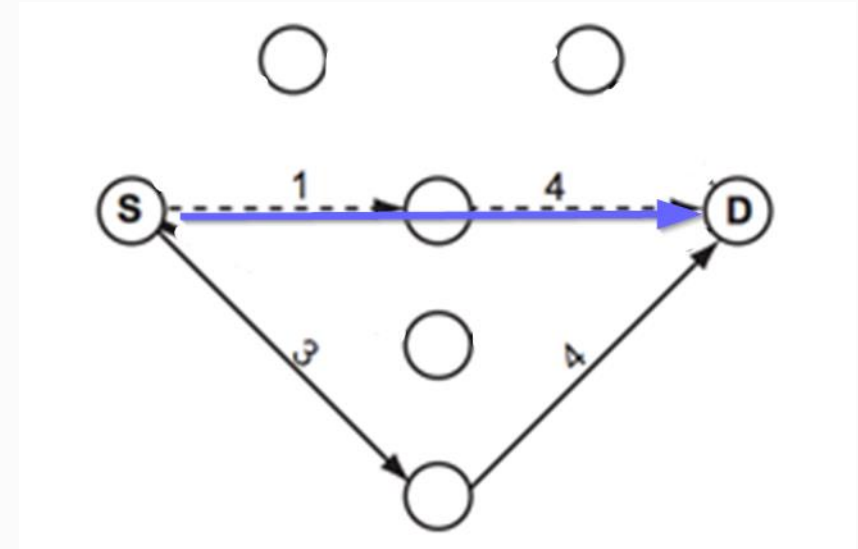
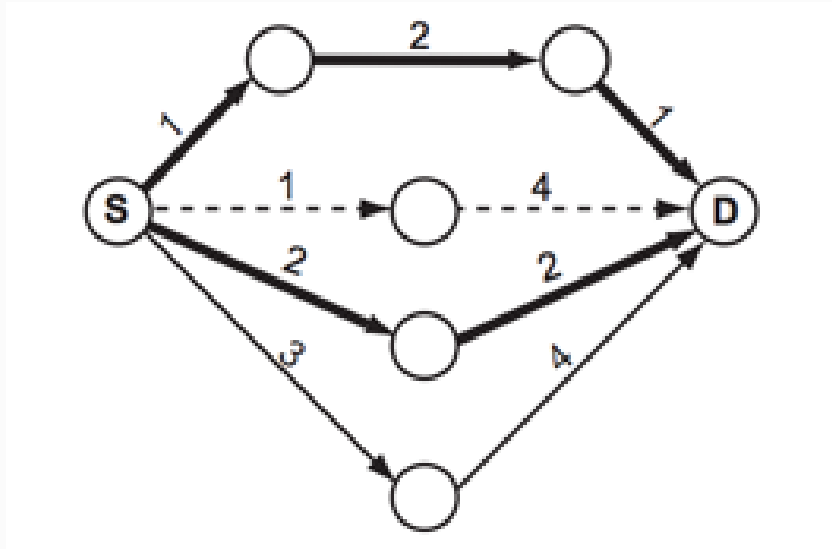


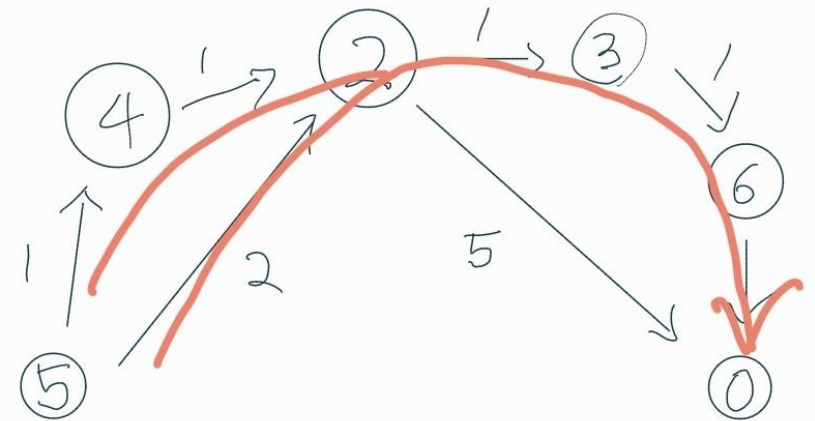
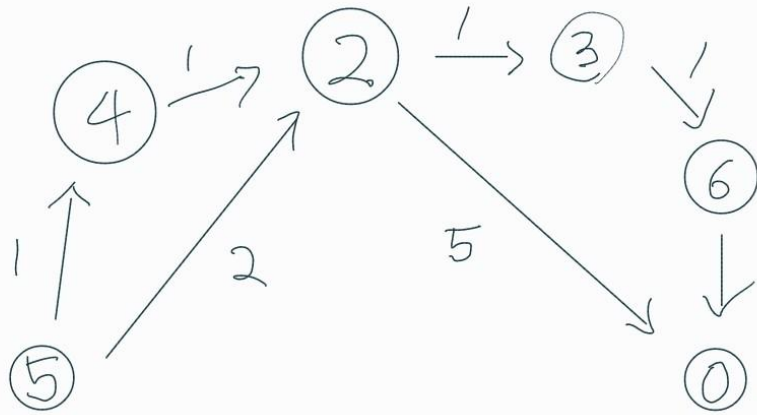
# 0. 거의 최단 경로

<https://www.acmicpc.net/problem/5719>



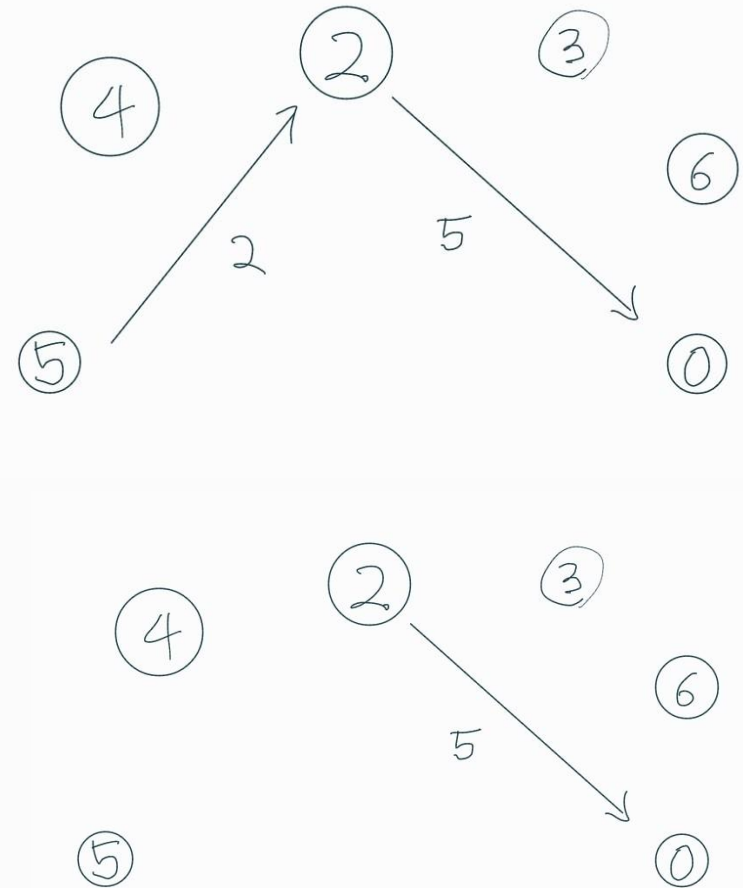
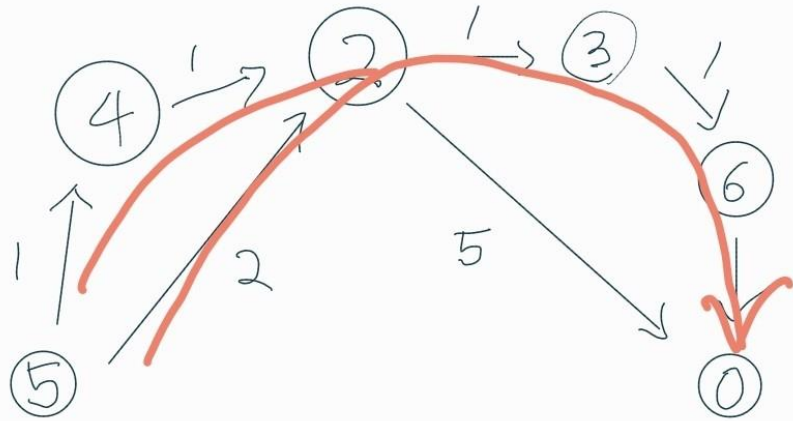
# 0. 거의 최단 경로

<https://www.acmicpc.net/problem/5719>



# 0. 거의 최단 경로

<https://www.acmicpc.net/problem/5719>



# DP

알고레인저 블루

# CONTENTS



## 001 DP 정의

- 분할 정복과 차이
- Memoization



## 002 DP 기초

- 피보나치
- 123 더하기
- 평범한 배낭



## 003 DP 응용

- 구간 합
- SKK 문자열



## 004 여러가지 문제

# 1. DP

Dynamic Programming



특정 범위까지의 값을 구하기 위해서, 그것과 다른 범위까지의 값을 이용하여 구하는 알고리즘



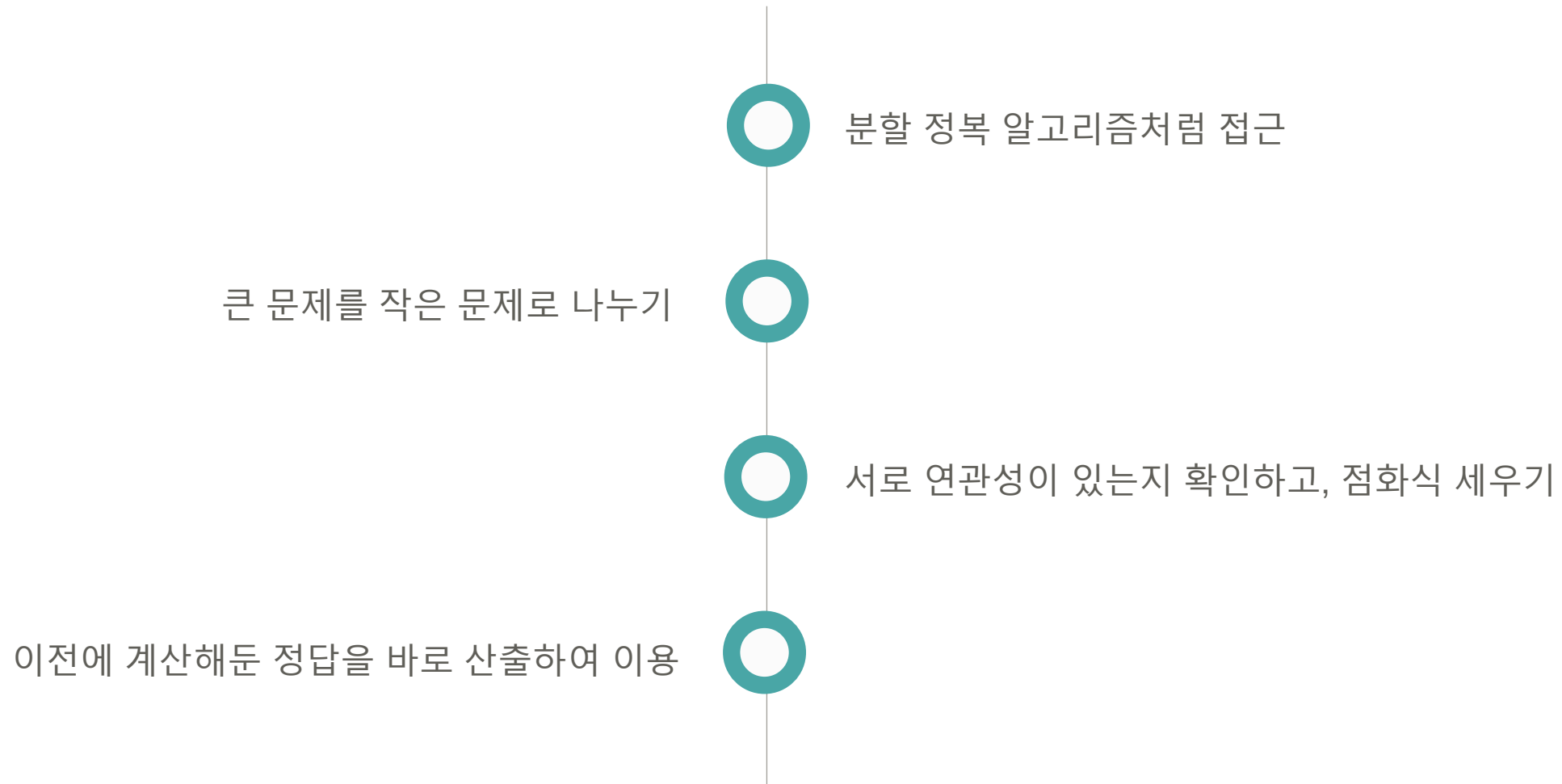
기억하며 풀기. => 이전 계산 내용을 기억했다가 다음에 써먹기.



**Memoization**임. **Memorization**이 아니다



피보나치, 0-1 배낭 문제, 가장 긴 증가 수열 문제, 플로이드-워셜



# Part 1.

---

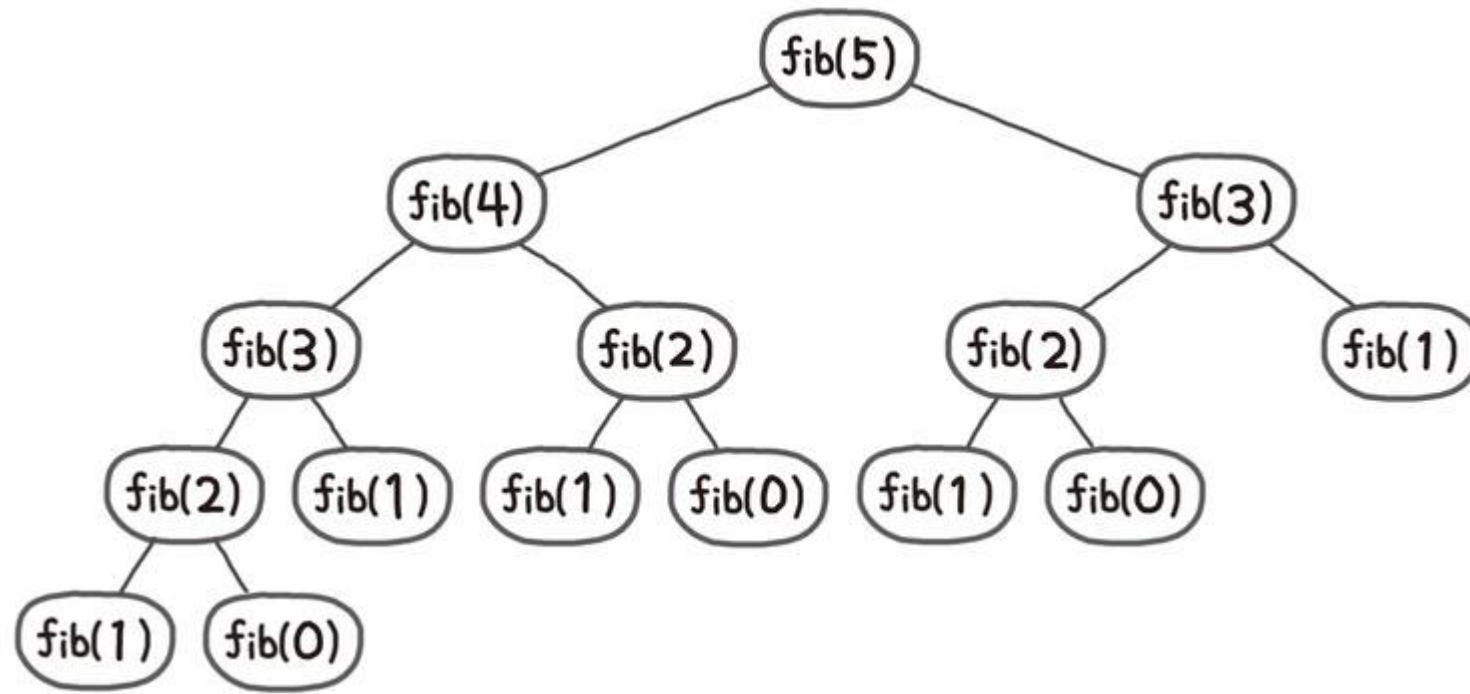
## DP 기본 문제





## 2-1. 피보나치

Dynamic Programming

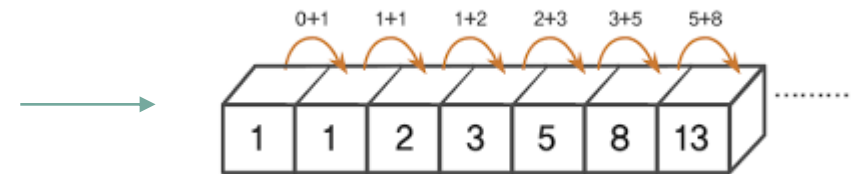


$$\text{Fibo}(5) = \text{Fibo}(4) + \text{Fibo}(3)$$

$$\text{Fibo}(4) = \text{Fibo}(3) + \text{Fibo}(2)$$

$$\text{Fibo}(3) = \text{Fibo}(2) + \text{Fibo}(1)$$

$$\longrightarrow \text{Fibo}(n) = \text{Fibo}(n-1) + \text{Fibo}(n-2)$$



## 2-2. 1, 2, 3 더하기

Dynamic Programming

정수 4를 1,2,3의 합으로 나타내는 방법은 총 7가지가 있다. 합을 나타낼 때는 수를 1개 이상 사용해야 한다.

- 1+1+1+1
- 1+1+2
- 1+2+1
- 2+1+1
- 2+2
- 1+3
- 3+1

=> 정수 n이 주어졌을 때, n을 1, 2, 3의 합으로 나타내는 방법의 수를 구하는 프로그램을 작성하시오.

1,2,3을 이용해 정수 4를 만들 수 있는 경우의 수

$\Rightarrow$  (3을 만드는 경우) + 1

(1+1+1) +1

(1+2) +1

(2+1) +1

(3) +1

$\Rightarrow$  (2을 만드는 경우의 수) + 2

(1+1)+2

(2)+2

(1을 만드는 경우의 수) + 3

$\Rightarrow$  1+3

## 2-3. 평범한 배낭

Dynamic Programming

가방에 최대 무게  $K$  까지 담을 수 있다.  
배낭에 넣을 수 있는 물건들의 가치의 최댓값을 알려주자.

물건 번호	1	2	3	4
무게 Weight	5	4	6	3
가치 Value	10	40	30	50

물건 번호	1	2	3	4
Weight	5	4	6	3
Value	10	40	30	50

K[i, w]		w										
i	고려 물건들	0	1	2	3	4	5	6	7	8	9	10
0	{ }	0	0	0	0	0	0	0	0	0	0	0
1	{1}	0	0	0	0	0	10	10	10	10	10	10
2	{1,2}	0	0	0	0							
	...	.										

물건 번호	1	2	3	4
Weight	5	4	6	3
Value	10	40	30	50

K[i, w]		w										
i	고려 물건들	0	1	2	3	4	5	6	7	8	9	10
0	{ }	0	0	0	0	0	0	0	0	0	0	0
1	{1}	0	0	0	0	0	10	10	10	10	10	10
2	{1,2}	0	0	0	0	40	40					
	...	.										

물건 번호	1	2	3	4
Weight	5	4	6	3
Value	10	40	30	50

K[i, w]		w										
i	고려 물건들	0	1	2	3	4	5	6	7	8	9	10
0	{ }	0	0	0	0	0	0	0	0	0	0	0
1	{1}	0	0	0	0	0	10	10	10	10	10	10
2	{1,2}	0	0	0	0	40	40	40	40	40	50	50
	...	.										



# Part 2.

---

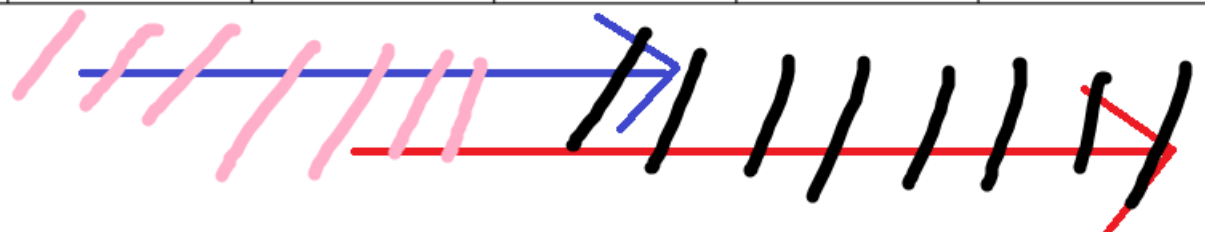
## DP 응용



## 2.1 누적 합

Dynamic Programming

	1	2	3	4	5
배열	7	6	3	2	1
누적합	7	13	16	18	19



특정 범위의 합을 빠르게 구하는 데 사용 할 수 있다.

$$\text{sum}[i] = \text{sum}[i-1] + \text{arr}[i]$$

## 2.1 누적 합

Dynamic Programming

	1	2	3	4	5
배열	7	6	3	2	1
누적합	7	13	16	18	19

배열 2~4까지의 합 =  $\text{sum}[4] - \text{sum}[1]$

$$6 + 3 + 2 = 18 - 7 = 11$$

## 2.1 누적 합

Dynamic Programming

	a(i,j)	1	2	3	4	5	6	
	1	0	1	1	0	0	1	
	2	0	0	0	0	0	1	
	3	1	0	1	0	1	1	
	S(i,j)	1	2	3	4	5	6	
	1	0	1	2	2	2	3	
	2	0	1	2	2	2	4	
	3	1	2	4	4	5	8	

## 2.1 누적 합

Dynamic Programming

a(i,j)	1	2	3	4	5	6
1	0	1	1	0	0	1
2	0	0	0	0	0	1
3	1	0	1	0	1	1

S(i,j)	1	2	3	4	5	6
1	0	1	2	2	2	3
2	0	1	2	2	2	4
3	1	2	4	4	5	8

(2,2)에서 (3,3)까지의 합

$$= \text{sum}[3][3] - \text{sum}[1][3] - \text{sum}[3][1] + \text{sum}[1][1]$$

## 2.2 SKK 문자열

<https://www.acmicpc.net/problem/24525>

K의 개수가 S의 개수의 정확히 2배인 문자열을 SKK 문자열이라고 한다.  
SKK문자열을 S,K 말고 다른 문자열도 포함할 수 있다.

문자열이 주어질 때, 부분 문자열을 가장 긴 SKK 문자열을 찾아라.

LUKESKYWALKER → LUKESKYWALKER

## 2.2 SKK 문자열

<https://www.acmicpc.net/problem/24525>

DP 배열을 만드는데,  
S가 나오면 +2, K가 나오면 -1을 해보자.

S	A	K	S	K	K	U	K
2	2	1	3	2	1	1	0

## 2.2 SKK 문자열

<https://www.acmicpc.net/problem/24525>

S	A	K	S	K	K	U	K
2	2	1	3	2	1	1	0

0이 된다 => K의 개수가 2의 개수 \* 2

SAKSKKUK는 SKK 문자열이다.



## 2.2 SKK 문자열

<https://www.acmicpc.net/problem/24525>

1	2	3	4	5	6	7	8
S	A	K	S	K	K	U	K
2	2	1	3	2	1	1	0

KSK는 SKK 문자열이다.

1	2	3	4	5	6	7	8
S	A	K	S	K	K	U	K
2	2	1	3	2	1	1	0

AKSK는 SKK 문자열이다.

## 2.3 한계?

Dynamic Programming

	1	2	3	4	5
배열	7	6	3	2	1
누적합	7	13	16	18	19

Arr[1]을 7에서 20으로 바꾸면?



배열	20	6	3	2	1
누적합	20	26	29	31	32

# Part 3.

---

## 문제 풀자



<https://programmers.co.kr/learn/courses/30/lessons/42895>