

# Binary Search 활용.

사용하는 경우.

1. 임의의 수가 문제를 해결 할 수 있는 경우인지  $O(n)$ 의 시간 복잡도 내에 해결 가능하다.
- 2-a) 임의의 수가 해당 문제를 해결 가능한 정합의 원소일 때,  
해결 가능한 경우, 그 아래의 수는 해당 문제를 해결 할 수 있다
- 2-b) 임의의 수가 해당 문제를 해결 가능한 정합의 원소에 속하지  
않을 때, 그 이상의 수는 모두 해당 문제를 해결 할 수 있다

Assume that  $S$  is a set of all possible solution and  
for  $x_i$  where  $a \leq i \leq b$ ,  $x_i$  is candidate o'  $S$ .

if  $x_i \in S$ ,  $x_j \in S$  where  
 $a \leq j \leq b$ .

if  $x_i \notin S$ ,  $x_j \in S$  where  $a \leq j \leq b$ .

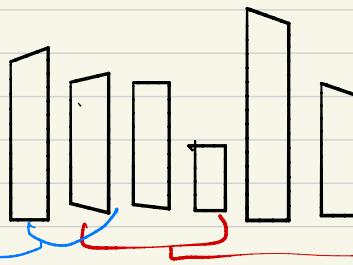
# Binary Search 활용 예제

kakao 2020 winter intern coding test 1st - 5th.

- 숫자가 주어졌을 때 해당 값이 정답인지  $O(n)$  안에 확인 가능하다.

총  $n$  명의 학생이 다리를 통과 가능한지 확인해보자. 이를 위해선 정길다리를 높이가  $1$  미만인 둘의 연속으로  $k$  개 이상 존재하는 경우가 없어야 한다.

(ex)



위 그림에서  $k=3$ ,  $n$ 은 4일 경우 해당 부분의 높이가 3이하인 연속 3개에 해당한다. 이 때문에 4명의 학생이 정길다리를 건널 수 없다.

위 그림에서  $n=5$ ,  $k=5$ 일 경우 높이가 5미만인 연속한 둘의 최대 개수는 4이므로 5명이 정길다리를 통과할 수 있다.

연속하는 둘의 최대 개수를 세는 연산은 처음부터 마지막까지 둘의 높이를 조회후 비교하는  $O(N)$  연산이다.

2. 원의의 수가 수 문제를 해결  
(을) 가능할 때 나머지 정  
형을 제한할 수 있다.

7명이 징검다리를 건널 수 있을 때 8명 이하의 인원  
이 징검다리를 건너는 것은 자명하다

1명이 징검다리를 건널 수 없을 때 그 이상의 인원이 징  
검다리를 건널 수 없음을 자명하다.

# Unionfind.

사용 조건.	임의의 수가 주어졌을 때, 수가 속한 집합에 따라 값이 결 정되는 경우에 사용한다.
kakao 2020 winter intern Coding test 1st - 4th.	
Point 1)	연속된 봉들이 예약된 경우, 이들이 input으로 들어왔을 때 같은 봉을 return해야한다. 즉 연속한 수는 모두 같은 집합에 속함으로 union-find를 사용 가능하다. ex) 1 2 4 5 7 10 11 12 13
point 2)	일반적인 Union-find algorithm은 array를 통해 hash map 을 구현하지만 해당 문제는 정원의 수가 최대 $10^5$ 이므로 해당 언어에 내재하는 hash를 사용한다. ex) unordered_map in C++.

# Back tracking. (Branch cutting)

사용하는 경우.	1) 모든 경우의 수를 확인해야하는 경우. 2) 후보 중 하나를 선택할 때마다 나머지 후보를 배제하는 경우. 3) greedy / dynamic algorithm으로 해결 불가능한 경우 ( 대부분 가능하다).
Enumeration	Subset Sum: 주어진 수의 집합으로 target sum을 만들 수 있는 경우를 구하시오.
Optimization.	Maximum Score words formed by letters. (leetcode 1255).
Decision	m Coloring problem: m개의 색이 주어졌을 때 graph의 vertices를 연결한 것은 같은 색을 갖지 않도록 하는 것이 가능한지 판斷하시오.