

## 07. 이분 탐색

## 이분 탐색이란?

데이터가 정렬되어 있는 배열에서 사용자가 원하는 값을 찾아내는 알고리즘

모든 값을 순회하지 않고도 답을 찾아낼 수 있는 방법

## 이분 탐색이란?

11?

### 1. 초기값 설정

- 가능한 정답의 최솟값 : left
- 가능한 정답의 최댓값 : right

left

right

1	3	5	7	9	11	13
---	---	---	---	---	----	----

## 이분 탐색이란?

11?

2. 중간값(mid) 설정  $\rightarrow (left + right) / 2$

left		mid			right	
1	3	5	7	9	11	13

## 이분 탐색이란?

11?

### 3. 중간 점검

mid 가 가리키는 값보다 작을 때  $\rightarrow \text{left} = \text{mid} + 1$

mid 가 가리키는 값보다 클 때  $\rightarrow \text{right} = \text{mid} - 1$

			mid	left	right	
1	3	5	7	9	11	13

## 이분 탐색이란?

11?

2. 중간값(mid) 설정  $\rightarrow (left + right) / 2$

				left	mid	right
1	3	5	7	9	11	13

## 이분 탐색이란?

11?

3. 중간 점검

-> 검거!

				left	mid	right
1	3	5	7	9	11	13

## 이분 탐색이란?

예외. 배열에 원하는 값이 없을 때

					mid	left right
1	3	5	7	9	10	13



## 이분 탐색이란?

이런 문제 보셨나요?..



## 파라메트릭 서치

최적화 문제를 결정 문제로 바꾸어 푸는 것

이분 탐색과 매우 유사

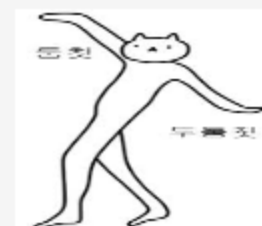
## 파라메트릭 서치

최적화 문제 : kau 클럽에 갈 수 있는 사람들 중 키가 가장 작은 사람 찾기

결정 문제 : 너 클럽에 갈 수 있니? "네" or "아니오"



kau 클럽



## 파라메트릭 서치

풀이 방법

1. 초기값 설정 (left, right)
2. 정답 하나 결정 후(mid) 조건에 맞는지 계산하는 함수(calc)
3. 조건에 따라 범위 수정 후 2번으로 되돌아가기

## 기본 코드

```
// 초기화
int left = 0, right = n;
while (left <= right)
{
    int mid = (left + right) / 2;
    // 계산값이 조건을 충족한다면 정답 초기화
    if (calc(mid)) {
        ans = mid;
        left = mid + 1;
    }
    // 아니라면 더 작은 값을 선택
    else right = mid - 1;
}
```

## 시간 복잡도

한 번의 계산마다 선택지가 절반씩 줄어드므로  $O(\log N)$

단, 배열이 정렬되지 않은 경우 정렬하기 위해  $O(N \log N)$  인 경우도 있다!

## 랜선 자르기

BOJ 1654번

k개의 랜선이 있다.

랜선을 동일한 길이로 잘라 총 N개 이상의 랜선을 만들어야 하는 문제

300cm 랜선에서 140cm 랜선 두 개를 자르면 나머지 20cm 랜선은 버린다.

조건을 만족하는 최대 랜선 길이는?

## 랜선 자르기

BOJ 1654번

예제 [802, 743, 457, 539] 에서 11개의 랜선을 찾는 경우

150cm 기준으로 랜선을 자르면?

$$802/150 = 5$$

$$743/150 = 4$$

$$457/150 = 3$$

$$539/150 = 3$$

$$5 + 4 + 3 + 3 = 15$$



## 랜선 자르기

BOJ 1654번

예제 [802, 743, 457, 539] 에서 11개의 랜선을 찾는 경우

150cm 기준으로 랜선을 자르면?

$$802/150 = 5$$

$$743/150 = 4$$

$$457/150 = 3$$

$$539/150 = 3$$

$$5 + 4 + 3 + 3 = 15$$

분모가 작아지면, 랜선의 개수는 점점 많아진다.

따라서 150cm 보다 작은 경우는 구해볼 필요가 없다.

## 랜선 자르기

BOJ 1654번

예제 [802, 743, 457, 539] 에서 11개의 랜선을 찾는 경우

200cm 기준으로 랜선을 자르면?

$$802/200 = 4$$

$$743/200 = 3$$

$$457/200 = 2$$

$$539/200 = 2$$

$$4 + 3 + 2 + 2 = 11$$

11개 이상 자를 수 있다면, 최대한 크게 만들어야 한다.

-> 그 기준으로 계속 이분 탐색을 이용한다!

## 랜선 자르기

BOJ 1654번

풀이 과정

1.  $left = 0$ ,  $right$  = 랜선의 최대 길이
2. 중간값( $mid$ ) 길이로 랜선이 몇개 나오는 지 계산한다. (위 예시 참고)
3. 중간값으로 랜선을  $N$ 개 이상 만들 수 있다면  $left = mid + 1$   
없다면  $right = mid - 1$

## 기타 레슨

BOJ 2343번

N개의 레슨을 블루레이로 만들어 팔려고 한다.

총 M개의 블루레이를 이용할 때, 가능한 블루레이의 크기 중 최소값은?

단, 레슨의 순서가 바뀌면 안된다!

## 기타 레슨

BOJ 2343번

문제에서 주어지는 힌트!

1. 가능한 **블루레이 크기**의 최댓값

-> 레슨의 수 최대 10만 x 개당 길이 1만 = 10억

2. 순서대로

## 기타 레슨

BOJ 2343번

$N = 9, M = 3$

[1, 2, 3, 4, 5, 6, 7, 8, 9]

## 기타 레슨

BOJ 2343번

$N = 9, M = 3$

[1, 2, 3, 4, 5, 6, 7, 8, 9]

만약 블루레이의 크기  $x = 15$ ?

[1, 2, 3, 4, 5], [6, 7], [8], [9] 총 4개!

-> 크기를 늘린다 -> 배열에 들어가는 원소가 커진다 -> 블루레이 개수가 줄어든다

## 기타 레슨

BOJ 2343번

$N = 9, M = 3$

[1, 2, 3, 4, 5, 6, 7, 8, 9]

만약 블루레이의 크기  $x = 25$ ?

[1, 2, 3, 4, 5, 6], [7, 8, 9] 총 2개!

아직 하나를 안썬기 때문에 최적이 아닐 수 있다!



# 기타 레슨

BOJ 2343번

## 계산 코드

```
bool calc(int x)
{
    int cnt = 1; //블루레이의 개수
    int sum = 0; //현재까지 영상 길이의 합
    for (int i = 0; i < n; i++)
    {
        //하나의 영상이 블루레이 크기보다 크면 불가능
        if (v[i] > x) return false;
        sum += v[i];
        //하나를 더 담았는데 블루레이가 터진 경우 -> 추가
        if (sum > x) {
            sum = v[i];
            cnt++;
        }
    }
    //개수가 m보다 작거나 같을 때 -> 조건 만족
    if (cnt <= m) return true;
    else return false;
}
```

## 공유기 설치

BOJ 2110번

집  $N$ 개가 수직선 상에 위치해 있다. (좌표 서로 다르다)

공유기  $C$ 개 설치, 가능한 멀찍이

$C$ 개의 공유기를  $N$ 개의 집에 적당히 설치해서 가장 인접한 두 공유기 사이 거리를 최대로 만드는 문제

## 공유기 설치

BOJ 2110번

입력 조건

$N, C$  (2 ~ 20만)

$x_i$  ( $0 \leq x_i \leq 1,000,000,000$ )

-> 이 정도의 범위는 오히려 힌트가 될 수 있다!

## 공유기 설치

BOJ 2110번

우선, 문제에서 물어보는 공유기 사이 거리를 기준으로 잡는다.

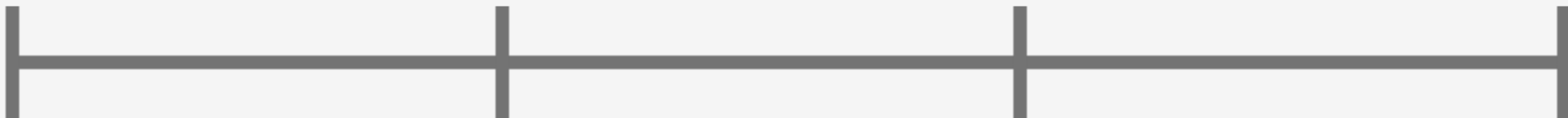
최대 거리로 공유기를 두려면 어떻게 해야 할까?



## 공유기 설치

BOJ 2110번

가장 왼쪽 집부터 공유기를 설치해야, 최대한 멀게 설치할 수 있다!



## 공유기 설치

BOJ 2110번

가장 인접한 두 공유기 사이 거리를  $mid$  라고 하고 공유기를 설치해보자

$mid = 2$



## 공유기 설치

BOJ 2110번

가장 인접한 두 공유기 사이 거리를 mid 라고 하고 공유기를 설치해보자

mid = 5



## 공유기 설치

BOJ 2110번

정리

1. 공유기 사이 거리를  $x$ 로 설정
2.  $x$ 를 기준으로 공유기 설치
3. 공유기를  $C$ 개 이상 설치할 수 있다  $\rightarrow$  여유가 있다!

따라서  $x$ 보다 짧은 거리는 볼 필요 없이 거리를 늘려 최댓값 확인!