

< N >
가
 $O(\log N)$

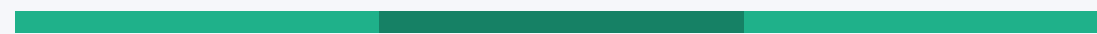
K

.

1

이분 탐색으로 정답 찾기

최백준 choi@startlink.io



이분 탐색으로 정답 찾기

이분 탐색으로 정답 찾기

3

Binary Search

- 정답을 구하는 것은 어렵지만
- X가 가능한지 아닌지를 알아내는 것은 쉬운 문제들이 있다 // NP

이분 탐색으로 정답 찾기

4

Binary Search

- 정답을 구하는 문제
- A에서 B까지 가는 가장 빠른 시간을 구하는 것
- 가능한지 살펴보는 문제
 - A에서 B까지 X라는 시간으로 이동할 수 있나?



이분 탐색으로 정답 찾기

5

Binary Search

- 정답을 구하는 문제는 가능여부를 판별하는 문제로 바꿀 수 있다.
- A에서 B까지 가는 가장 빠른 시간이 M인 경우에
 - M보다 빠른 시간은 모두 불가능
 - M보다 큰 시간은 모두 가능

이분 탐색으로 정답 찾기

Binary Search

- 가능한지 아닌지를 알아보는 문제도 정답을 구하는 문제로 바꿀 수 있다 //

- A에서 B까지 1라는 시간으로 이동할 수 있나?
- A에서 B까지 2라는 시간으로 이동할 수 있나?
- A에서 B까지 3라는 시간으로 이동할 수 있나?
- A에서 B까지 4라는 시간으로 이동할 수 있나?
- A에서 B까지 5라는 시간으로 이동할 수 있나?
- A에서 B까지 6라는 시간으로 이동할 수 있나?
- A에서 B까지 7라는 시간으로 이동할 수 있나?
- ...

이분 탐색으로 정답 찾기

7

Binary Search

- 가능한지 아닌지를 알아보는 문제도 정답을 구하는 문제로 바꿀 수 있다
- A에서 B까지 1라는 시간으로 이동할 수 있나? 아니오
- A에서 B까지 2라는 시간으로 이동할 수 있나? 아니오
- A에서 B까지 3라는 시간으로 이동할 수 있나? 아니오
- A에서 B까지 4라는 시간으로 이동할 수 있나? 아니오
- **A에서 B까지 5라는 시간으로 이동할 수 있나? 예**
- A에서 B까지 6라는 시간으로 이동할 수 있나? 예
- A에서 B까지 7라는 시간으로 이동할 수 있나? 예
- A에서 B로 이동하는 가장 빠른 시간은 5이다.

이분 탐색으로 정답 찾기

8

Binary Search

- 어떤 기준 X를 가지고 Yes/No로 나누어지는 것만 정답을 찾을 수 있다.

!!

·
,

!

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- 랜선 K개가 있다
- 랜선을 같은 길이로 잘라서 N개를 만들어야 한다
- 300cm를 140cm로 자르면, 140cm 2개, 남은 20cm는 버린다
- 자른 것을 붙일 수 없다

랜선 자르기

10

<https://www.acmicpc.net/problem/1654>

- 예를 들어, 랜선이 4개가 있고, 11개를 만들어야 하는 경우
- 802, 743, 457, 539 인 경우
- 200으로 자르면 11개를 만들 수 있다
- $802/200 = 4$
- $743/200 = 3$
- $457/200 = 2$
- $539/200 = 2$
- $4+3+2+2 = 11$

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- 이 문제는 Binary Search를 이용해야 한다
- 만약, 길이 X 로 잘랐을 때, N 개 이상을 만들 수 있으면, X 를 크게 만들어 봐야 한다
- N 개 이상을 만들 수 없다면 X 를 작게 만들어야 한다

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- 예를 들어, 랜선이 4개가 있고, 11개를 만들어야 하는 경우
- 802, 743, 457, 539 인 경우
- 201로 자르는 경우
- $802/201 = 3$
- $743/201 = 3$
- $457/201 = 2$
- $539/201 = 2$
- $3+3+2+2 = 10$

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- 예를 들어, 랜선이 4개가 있고, 11개를 만들어야 하는 경우
- 802, 743, 457, 539 인 경우
- 199로 자르는 경우
- $802/199 = 4$
- $743/199 = 3$
- $457/199 = 2$
- $539/199 = 2$
- $4+3+2+2 = 11$

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- 예를 들어, 랜선이 4개가 있고, 11개를 만들어야 하는 경우
- 802, 743, 457, 539 인 경우
- 100으로 자르는 경우
- $802/199 = 8$
- $743/199 = 7$
- $457/199 = 4$
- $539/199 = 5$
- $8+7+4+5 = 24$

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- 랜선은 더 많이 만들어도 된다.

랜선 자르기

16

<https://www.acmicpc.net/problem/1654>

1. X라는 길이로 랜선을 만들어본다.
2. N개 이상 만들 수 있으면, X를 크게 바꾼다.
3. 만들 수 없으면 X를 작게 바꾼다.

랜선 자르기

<https://www.acmicpc.net/problem/1654>

- C/C++: <https://gist.github.com/Baekjoon/b0f6f757b2abb2983f09>
- Java: <https://gist.github.com/Baekjoon/c164259c72b1d39ce26d>

나무 자르기

H ->

H ->

18

<https://www.acmicpc.net/problem/2805>

- 상근이는 나무 M미터가 필요하다
- 절단기에 높이 H를 지정해야 한다
- 높이를 지정하면 톱날이 땅으로부터 H미터 위로 올라간다
- 한 줄에 연속해있는 나무를 모두 절단해버린다
- 높이가 H보다 큰 나무는 H 위의 부분이 잘릴 것이고, 낮은 나무는 잘리지 않을 것이다
- 적어도 M미터의 나무를 집에 가져가기 위해서 절단기에 설정할 수 있는 높이의 최대값

Left = 0 , Right = 20

나무 자르기

<https://www.acmicpc.net/problem/2805>

- 예를 들어 나무가 20, 15, 10, 17인 경우
- 절단기의 높이를 15로 하면
- 15, 15, 10, 15 가 되어 버리고
- 길이가 5인 나무와 2인 나무를 집에 들고 갈 수 있다

나무 자르기

20

<https://www.acmicpc.net/problem/2805>

1. X 라는 높이로 잘라본다. 나온 길이의 합 = C
2. C 가 M 보다 크거나 같으면 X 를 작게
3. 아니면 X 를 크게

나무 자르기

<https://www.acmicpc.net/problem/2805>

- C: <https://gist.github.com/Baekjoon/befaf4928a77148b6d3a>
- C++: <https://gist.github.com/Baekjoon/427d0d56b9ee4b0c3e4a>
- Java: <https://gist.github.com/Baekjoon/088cd44ca263bd32ce17>

공유기 설치

<https://www.acmicpc.net/problem/2110>

- 집 N개
- x_1, \dots, x_N
- 공유기 C개를 설치
- 가장 인접한 두 공유기 사이의 거리를 가능한 크게 //
- C개의 공유기를 N개의 집에 적당히 설치해서, 가장 인접한 두 공유기 사이의 거리를 최대

공유기 설치

<https://www.acmicpc.net/problem/2110>

- 집 N 개
- x_1, \dots, x_N
- 공유기 C 개를 설치
- 가장 인접한 두 공유기 사이의 거리를 가능한 크게
- C 개의 공유기를 N 개의 집에 적당히 설치해서, **가장 인접한 두 공유기 사이의 거리를 최대**

공유기 설치

<https://www.acmicpc.net/problem/2110>

- 가장 인접한 두 공유기 사이의 거리를 x 로 결정
- 이 때, 공유기를 설치했을 때, c 개 이상 설치할 수 있으면 가능!
- 가능한 경우면 x 를 크게
- 불가능하면 작게

공유기 설치

25

<https://www.acmicpc.net/problem/2110>

- C: <https://gist.github.com/Baekjoon/37661f1c6e34d9ba6079>
- C++: <https://gist.github.com/Baekjoon/a18b07450b8445e88594>
- Java: <https://gist.github.com/Baekjoon/379647e4eb53f4d64456>

구간 나누기 2

//

26

<https://www.acmicpc.net/problem/13397>

- 하나의 구간은 하나 이상의 연속된 수들로 이루어져 있다.
- 배열의 각 수는 모두 하나의 구간에 포함되어 있어야 한다.
- 구간의 점수 = 구간에 속한 수의 최대값 - 최소값
- 배열과 M이 주어졌을 때, 구간의 점수의 최대값의 최소값을 구하는 문제

구간 나누기 2

<https://www.acmicpc.net/problem/13397>

- 구간의 점수를 결정하고, 구간을 앞에서부터 차례대로 나눠보면 된다

구간 나누기 2

<https://www.acmicpc.net/problem/13397>

- C++: <https://gist.github.com/Baekjoon/2665d5d71b9d69f6ab8c7b9c845e988c>
- Java: <https://gist.github.com/Baekjoon/b0d15c448c12e2ab2826000426d2af3b>

중량제한

<https://www.acmicpc.net/problem/1939>

- N개의 섬과 M개의 다리로 이루어진 나라가 있다.
- 각 다리에는 중량제한이 있고, 중량제한을 초과하면 다리가 무너지게 된다
- 한 번의 이동에서 옮길 수 있는 물품들의 중량의 최대값을 구하는 문제

10 10 가 10 X
10 .
N M DFS/BFS !

중량제한

<https://www.acmicpc.net/problem/1939>

- 무엇을 결정해야 할까?

중량제한

<https://www.acmicpc.net/problem/1939>

- 최대 무게를 결정하고, 그 무게로 갈 수 있는지를 검사해본다

중량제한

<https://www.acmicpc.net/problem/1939>

- C/C++: <https://gist.github.com/Baekjoon/cc0deac0b4f18b2149ac>
- Java: <https://gist.github.com/Baekjoon/c54d6c65e8f5bf9148f9d508fc6a43a2>

배열에서 이동

<https://www.acmicpc.net/problem/1981>

- $n \times n$ 짜리의 배열이 하나 있다
- 이 배열의 (1, 1)에서 (n, n)까지 이동하려고 한다
- 이동할 때는 상, 하, 좌, 우의 네 인접한 칸으로만 이동할 수 있다.
- 이와 같이 이동하다 보면, 배열에서 몇 개의 수를 거쳐서 이동하게 된다
- 이동하기 위해 거쳐 간 수들 중 최대값과 최소값의 차이가 가장 작아지는 경우를 구하는 문제

배열에서 이동

<https://www.acmicpc.net/problem/1981>

- $n \times n$ 짜리의 배열이 하나 있다
- 이 배열의 (1, 1)에서 (n, n)까지 이동하려고 한다
- 이동할 때는 상, 하, 좌, 우의 네 인접한 칸으로만 이동할 수 있다.
- 이와 같이 이동하다 보면, 배열에서 몇 개의 수를 거쳐서 이동하게 된다
- 이동하기 위해 거쳐 간 수들 중 **최대값과 최소값의 차이**가 가장 작아지는 경우를 구하는 문제

배열에서 이동

35

<https://www.acmicpc.net/problem/1981>

- 차이 = 최대값 - 최소값
- 차이를 결정하고, 각각의 최소값에 대해서 최소~최대가 가능한지 조사해야 한다.

배열에서 이동

<https://www.acmicpc.net/problem/1981>

- C++: <https://gist.github.com/Baekjoon/bcbff7a160565595eb9b3634bd8f7d3f>
- Java: <https://gist.github.com/Baekjoon/427c12696448bbfe9459f648ab6333fb>

K번째 수

<https://www.acmicpc.net/problem/1300>

- 크기가 $N \times N$ 인 배열 A가 있다.
- $A[i][j] = i \times j$
- 이 배열에 들어있는 수를 모두 일차원 배열 B에 옮기고 정렬했다고 했을 때
- K번째 수는 무엇인가?
- $1 \leq N \leq 100,000$

K번째 수

<https://www.acmicpc.net/problem/1300>

- 실제로 배열을 만들 수 없다. // $10 \rightarrow 10 \times 10 = 100$
- 정렬했을 때, K번째 위치라는 것은 그 수보다 작은 수가 K개라는 것과 같은 의미이다.

K번째 수

39

<https://www.acmicpc.net/problem/1300>

- 8은 몇 번째에 있을까?

) K ?

Left = 1, Right = (n^2) (가 (n^2))


1	2	3	4	5
2	4	6	8	10
3	6	9	12	15
4	8	12	16	20
5	10	15	20	25

K번째 수

<https://www.acmicpc.net/problem/1300>

- 8은 몇 번째에 있을까?
- 14 ~ 15

(8 9 .)



1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

K번째 수

41

<https://www.acmicpc.net/problem/1300>

- 이분 탐색을 이용해서 정답을 구한다

K번째 수

<https://www.acmicpc.net/problem/1300>

- C++: <https://gist.github.com/Baekjoon/baadb577b0d513cd462115b69c1cfc3e>
- Java: <https://gist.github.com/Baekjoon/3c71e79f0d88ec08c6b2dbf44993ff37>

left, right
 (right가 left가
 left, right .)

```
while (left <= right) {
    long long mid = (left+right)/2;
    long long cnt = 0;
    for (long long i=1; i<=n; i++) {
        cnt += min(n, mid/i);
    }
    if (cnt >= k) {
        ans = mid;
        right = mid-1;
    } else {
        left = mid+1;
    }
}
```

```
while (left <= right) {
    long long mid = (left+right)/2;
    long long cnt = 0;
    for (long long i=1; i<=n; i++) {
        cnt += min(n, mid/i);
    }
    if (cnt >= k) {
        ans = mid;
        right = mid + 1;
    } else {
        left = mid - 1;
    }
}
```

놀이 공원

<https://www.acmicpc.net/problem/1561>

- N명이 있고, M종류의 1인승 놀이기구가 있다
- 줄의 마지막 아이가 타는 놀이기구의 번호를 구하는 문제

놀이 공원

<https://www.acmicpc.net/problem/1561>

- 무엇을 결정하고 문제를 풀어야 할까?

놀이 공원

<https://www.acmicpc.net/problem/1561>

- 문제를 바꿔서 풀어야 한다.
- x 분에 몇 번째 학생부터 몇 번째 학생이 놀이기구를 타는가?

놀이 공원

<https://www.acmicpc.net/problem/1561>

- 0분부터 12분까지 상황

	0	1	2	3	4	5	6	7	8	9	10	11	12
1	1	6	7	9	11	14	16	19	20	23	25	28	29
2	2		8		12		17		21		26		30
3	3			10			18			24			31
4	4				13				22				32
5	5					15					27		

놀이 공원

<https://www.acmicpc.net/problem/1561>

- 8분까지 탄 학생의 수는 몇 명인가?
- $5 + 8/1 + 8/2 + 8/3 + 8/4 + 8/5 = 22$

	0	1	2	3	4	5	6	7	8	9	10	11	12
1	1	6	7	9	11	14	16	19	20	23	25	28	29
2	2		8		12		17		21		26		30
3	3			10			18			24			31
4	4				13				22				32
5	5					15					27		

놀이 공원

48

<https://www.acmicpc.net/problem/1561>

- 8분에 탄 학생의 수는 몇 명인가?
- 3명 ($8\%1 == 0$, $8\%2 == 0$, $8\%4 == 0$)

	0	1	2	3	4	5	6	7	8	9	10	11	12
1	1	6	7	9	11	14	16	19	20	23	25	28	29
2	2		8		12		17		21		26		30
3	3			10			18			24			31
4	4				13				22				32
5	5					15					27		

놀이 공원

<https://www.acmicpc.net/problem/1561>

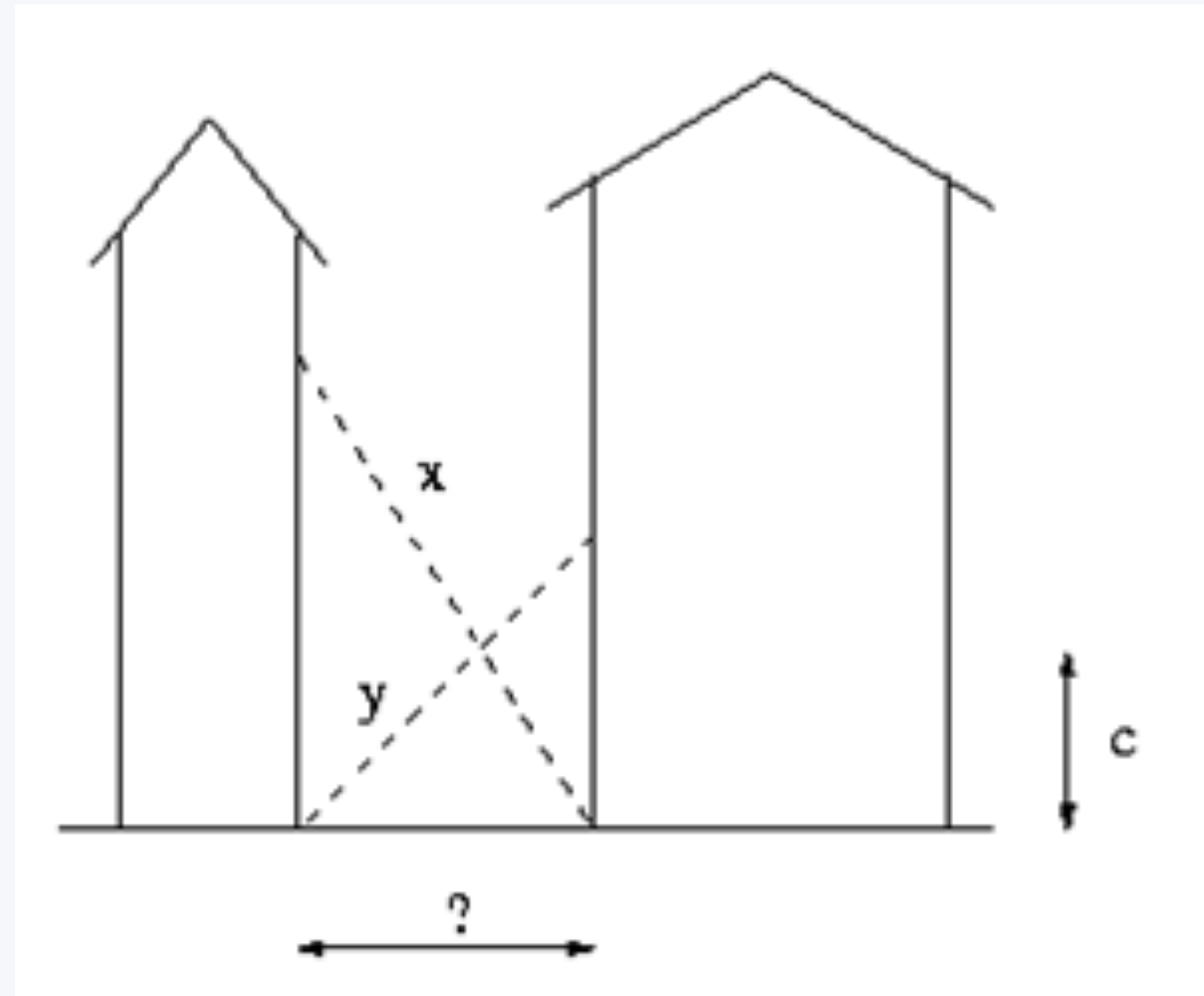
- C/C++: <https://gist.github.com/Baekjoon/a9ecc5eda03baa47ed17>
- Java: <https://gist.github.com/Baekjoon/e54f26f2a02372137e1173bc0e452934>

사다리

50

<https://www.acmicpc.net/problem/2022>

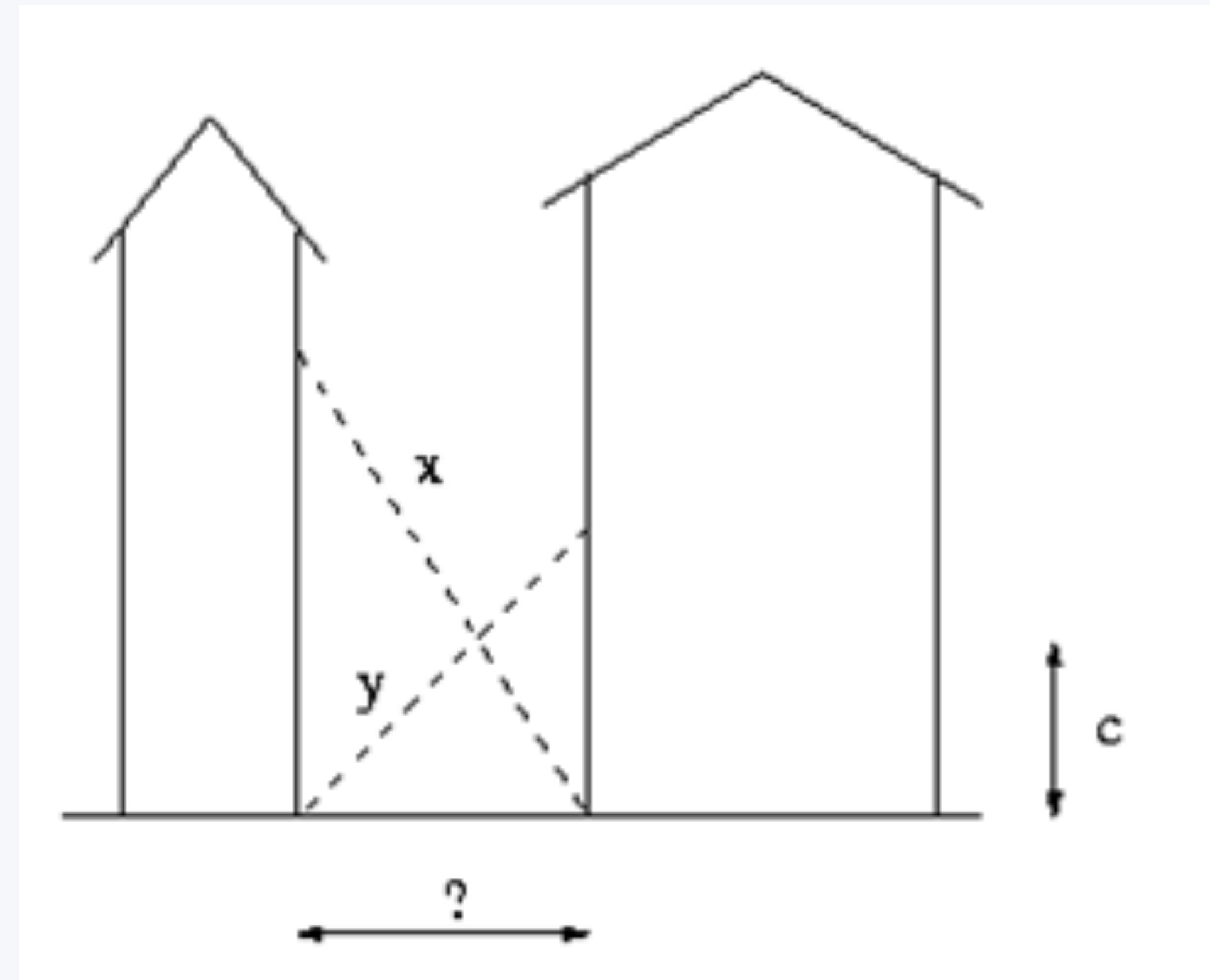
- x, y, c 가 주어졌을 때, 두 빌딩 사이의 떨어진 거리를 구하는 문제



사다리

<https://www.acmicpc.net/problem/2022>

- $h1 = \sqrt{x^2 - d^2}$
- $h2 = \sqrt{y^2 - d^2}$

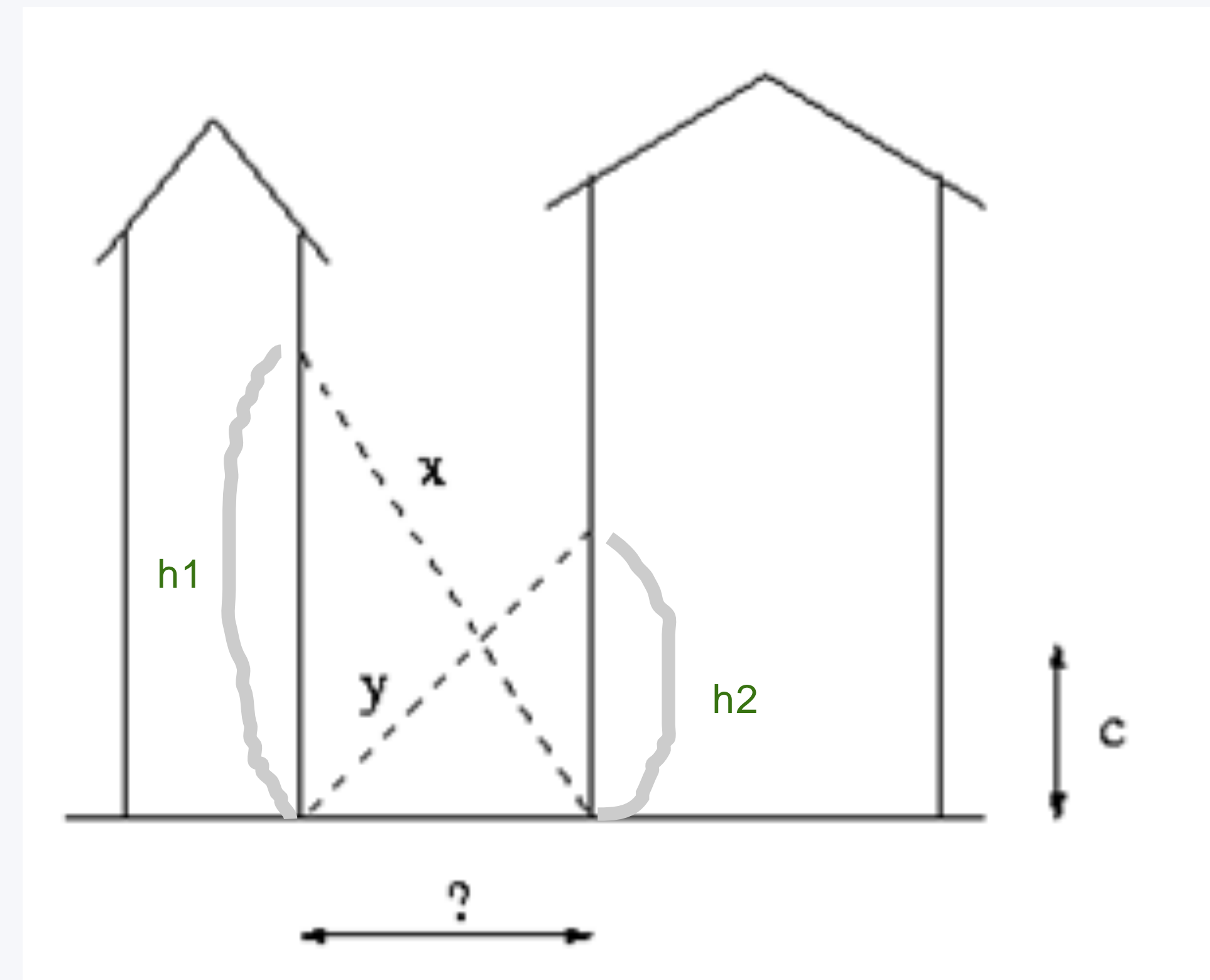


사다리

52

<https://www.acmicpc.net/problem/2022>

- 삼각형의 닮음을 이용하면
- $h_2/d = c/d_1$
- $h_1/d = c/d_2$
- $d = d_1 + d_2$
- $1/h_1 + 1/h_2 = 1/c$
- $c = h_1 * h_2 / (h_1 + h_2)$



사다리

<https://www.acmicpc.net/problem/2022>

- 실수에서 이분 탐색을 수행하는 것이기 때문에, 달라져야 하는 부분이 있다.
- `while (left <= right)` 와 같은 표현의 사용이 불가능하다
- `left = mid + 1` 불가능 (`mid` 와 `mid + 1` 사이에 정답이 있을 수 있기 때문)
- `right = mid - 1` 불가능 (`mid - 1` 와 `mid` 사이에 정답이 있을 수 있기 때문)

사다리

<https://www.acmicpc.net/problem/2022>

- 실수에서 이분 탐색을 수행하는 것이기 때문에, 달라져야 하는 부분이 있다.
- `while (left <= right)` 와 같은 표현의 사용이 불가능하다
- `left = mid + 1` \Rightarrow `left = mid`
- `right = mid - 1` \Rightarrow `right = mid`

사다리

<https://www.acmicpc.net/problem/2022>

- `while (left <= right)` 와 같은 표현의 사용이 불가능하다

- 대신 아래와 같은 표현 중 하나를 사용한다

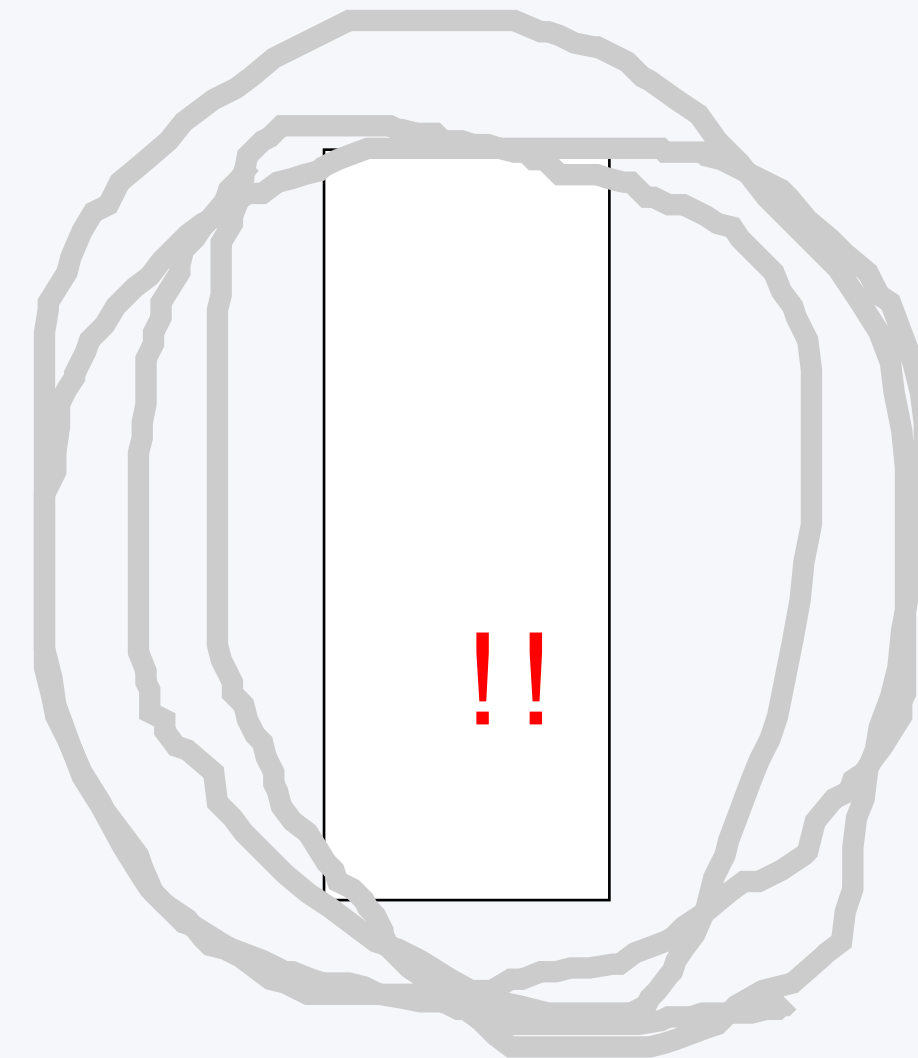
- `for (int k=0; k<100000; k++)`

- `while (abs(right-left) > 1e-6)`

가

6

/



사다리

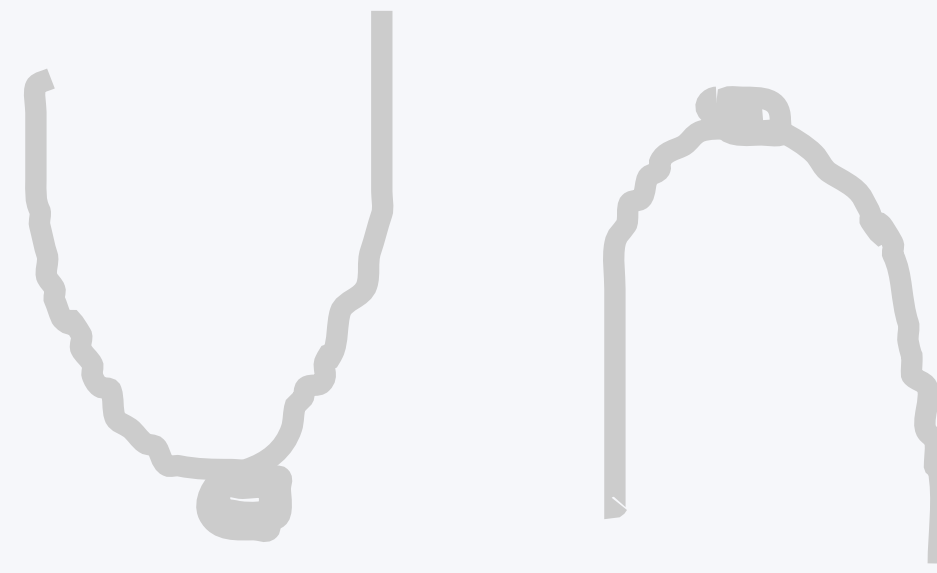
<https://www.acmicpc.net/problem/2022>

- C/C++: <https://gist.github.com/Baekjoon/5a2bd2a33eb855a0144ee38708431fe5>
- C/C++: <https://gist.github.com/Baekjoon/3472fd75d793be0140a99a232593d1d8>
- Java: <https://gist.github.com/Baekjoon/929e619f895ec1eb66526b7605be30f0>
- Java: <https://gist.github.com/Baekjoon/377a8453452f93764f189d7cd2510ec5>

삼분 탐색

삼분 탐색

Ternary Search



58

- 최소값 또는 최대값이 하나인 함수 (Unimodal function)에서 최소/최대값을 찾는 방법
- 이분 탐색과 비슷하지만, 삼등분을 한다.

삼분 탐색

Ternary Search

- 최대값을 찾는 경우 아래와 같이 동작한다.

```
while (true) {  
    if (abs(right-left) < 1e-9) return (left+right)/2;  
    double m1 = left + (right-left)/3;  
    double m2 = right - (right-left)/3;  
    if (f(m1) < f(m2)) {  
        left = m1;  
    } else {  
        right = m2;  
    }  
}
```

선분과 점

<https://www.acmicpc.net/problem/11664>

- 3차원 좌표 평면 위에 선분 하나와 점 하나가 있을 때
- 선분과 점 사이의 거리의 최소값을 구하는 문제

선분과 점

<https://www.acmicpc.net/problem/11664>

- 선분과 점 사이의 거리는 가까워졌다가 멀어진다.
- 즉, 삼분 탐색을 이용해 최소값을 구하는 문제이다.

선분과 점

<https://www.acmicpc.net/problem/11664>

- C++: <https://gist.github.com/Baekjoon/f0e5ab32c81ef5a47a482d0c1d69fbf1>
- Java: <https://gist.github.com/Baekjoon/8d8abc3e98055172bead0ef346608881>