# 시간복잡도

# 이분탐색

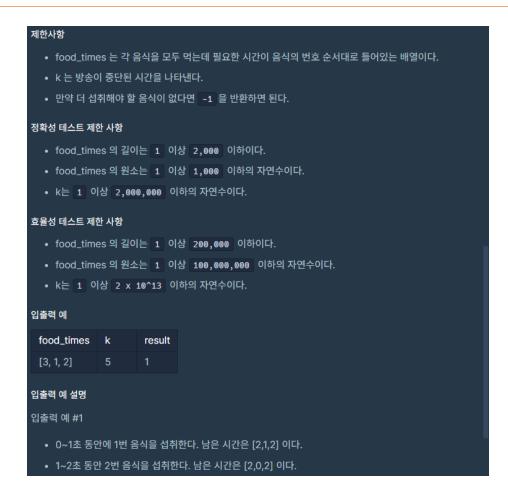
발표자 : 이윤수(lys9546)



https://www.acmicpc.net/problem/2751

제출 번호	아이디	문제	결과	메모리	시간	언어	코드 길이	제출한 시간
12895054	lys9546	<b>5</b> 2751	시간 초과			C99 / 수정	881 B	3년 전
12883531	lys9546	<b>5</b> 2751	맞았습니다!!	12428 KB	484 ms	C99 / 수정	1098 B	3년 전
12883505	lys9546	<b>5</b> 2751	틀렸습니다			C99 / 수정	1098 B	3년 전
12858303	lys9546	<b>5</b> 2751	맞았습니다!!	4904 KB	528 ms	C99 / 수정	1555 B	3년 전
12858294	lys9546	<b>5</b> 2751	틀렸습니다			C99 / 수정	1555 B	3년 전
12858277	lys9546	<b>5</b> 2751	틀렸습니다			C99 / 수정	1555 B	3년 전
12858147	lys9546	<b>5</b> 2751	시간 초과			C99 / 수정	1543 B	3년 전





https://school.programmers.co.kr/learn/courses/30/lessons/42891

BruteForcing

DP Greedy DFS, BFS BinarySearch Segment Tree LCA Network Flow

1초 1억

#### Big 0 Notation

```
for (int i = 0; i < N; i++)
{
    for (int j = 0; j < N; j++)
    {
        N = 100 10000
        N = 1000 1000000</pre>
```



https://www.acmicpc.net/problem/2751

정렬 종류		공간 복잡도			
02 5T	평균(Average)	최선(Best)	최악(Worst)	<u> </u>	
선택 정렬	O(n2)	O(n2)	O(n2)	O(n2)	
버블 정렬	O(n2)	O(n2)	O(n2)	O(n)	
삽입 정렬	O(n2)	O(n)	O(n2)	O(n2)	
합병 정렬	합병 정렬 O(n×log n)		O(n×log n)	O(n×log n)	
퀵 정렬	O(n×log n)	O(n×log n)	O(n2)	O(n×log n)	
힙 정렬	O(n×log n)	O(n×log n)	O(n×log n)	O(n×log n)	
쉘 정렬 O(N^1.25)		O(N^1.25)	O(N^1.25)	O(n)	
기수 정렬	O(dn)	O(dn)	O(dn)		

- for 문

- 재귀

대중적인 알고리즘의 시간복잡도는 알려져있음

+ 본인이 사용하는 내장함수의 시간복잡도는 알아야함

O(N+E)

https://currygamedev.tistory.com/10

#### list

The Average Case assumes parameters generated uniformly at random.

Internally, a list is represented as an array; the largest costs come from grainserting or deleting somewhere near the beginning (because everything a collections.deque instead.

Operation	Average Case	Amortized Worst Case		
Сору	O(n)	O(n)		
Append[1]	O(1)	O(1)		
Pop last	O(1)	O(1)		
Pop intermediate[2]	O(n)	O(n)		
Insert	O(n)	O(n)		
Get Item	O(1)	O(1)		
Set Item	O(1)	O(1)		
Delete Item	O(n)	O(n)		
Iteration	O(n)	O(n)		
Get Slice	O(k)	O(k)		
Del Slice	O(n)	O(n)		
Set Slice	O(k+n)	O(k+n)		
Extend[1]	O(k)	O(k)		
Sort Sort	O(n log n)	O(n log n)		
Multiply	O(nk)	O(nk)		
x in s	O(n)			
min(s), max(s)	O(n)			
Get Length	O(1)	O(1)		

https://wiki.python.org/moin/TimeComplexity

- 1. 코드의 실행 속도
- 2. 문자열 처리
- 3. 언어의 난이도/ 자료의 양

https://blog.encrypted.gg/965 https://medium.com/@cookatrice/why-python-is-slow-looking-under-the-hood-7126baf936d7

```
/* C code */
int a = 1;
int b = 2;
int c = a + b;
```

#### C 덧셈

- 1. <int> 1을 a에 할당
- 2. <int> 2을 b에 할당
- 3. binary\_add<int, int>(a,b) 호출
- 4. 결과를 c에 할당

a

b

1

2

# python code

a = 1

b = 2

c = a + b

1a. a->PyObject\_HEAD->typecode 정수 설정

1b. a->val = 1 설정

2. b에 2를 할당

2a. b->PyObject\_HEAD->typecode 정수 설정

2b. b->val = 2 설정

3. binary\_add(a,b) 호출

3a. a->PyObject\_HEAD 에서 typecode 찾기

3b. a는 정수형; 값 a->val

3c. b->PyObject\_HEAD 에서 typecode 찾기

3d. b는 정수형; 값 b->val

3e. binary\_add<int, int>(a->val, b->val) 호출

3f. 정수형 결과값 result

4. 파이썬 개체 c 생성

4a. c->PyObject\_HEAD->typecode 정수 설정

4b. c->val에 result 설정

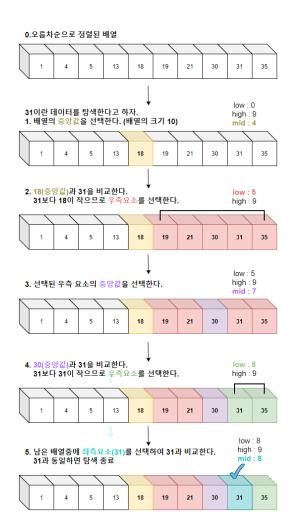
- 1. 코드의 실행 속도
- 2. 문자열 처리
- 3. 언어의 난이도/ 자료의 양

https://blog.encrypted.gg/965 https://medium.com/@cookatrice/why-python-is-slow-looking-under-the-hood-7126baf936d7

19

1 4 5 13 18 19 21 30 31 35

O(N)



https://www.acmicpc.net/blog/view/109

O(logN)

https://velog.io/@ming/%EC%9D%B4%EB%B6%84%ED%83%90%EC%83%89Binary-Search

https://www.acmicpc.net/problem/1072

BruteForcing

BinarySearch

#### 문제

김형택은 지금 몰래 Spider Solitaire(스파이더 카드놀이)를 하고 있다. 형택이는 이 게임을 이길 때도 있었지만, 질 때도 있었다. 누군가의 시선이 느껴진 형택이는 게임을 중단 하고 코딩을 하기 시작했다. 의심을 피했다고 생각한 형택이는 다시 게임을 켰다. 그 때 형택이는 잠시 코딩을 하는 사이에 자신의 게임 실력이 눈에 띄게 향상된 것을 알았다.

이제 형택이는 앞으로의 모든 게임에서 지지 않는다. 하지만, 형택이는 게임 기록을 삭제 할 수 없기 때문에, 자신의 못하던 예전 기록이 현재 자신의 엄청난 실력을 증명하지 못한다고 생각했다.

게임 기록은 다음과 같이 생겼다.

- 게임 횟수 : X
- 이긴 게임: Y (Z%)
- Z는 형택이의 승률이고, 소수점은 버린다. 예를 들어, X=53, Y=47이라면, Z=88이다.

X와 Y가 주어졌을 때, 형택이가 게임을 최소 몇 번 더 해야 Z가 변하는지 구하는 프로그램을 작성하시오.

#### 입력

각 줄에 정수 X와 Y가 주어진다.

#### 출력

첫째 줄에 형택이가 게임을 최소 몇 판 더 해야하는지 출력한다. 만약 Z가 절대 변하지 않는다면 -1을 출력한다.

#### 제한

- $1 \le X \le 1,000,000,000$
- 0 ≤ Y ≤ X

# Q & A