

기업 코딩테스트 극복하기

김민상 (a.k.a tony9402)

2022.11.05

발표자 소개

1 tony9402

<https://github.com/tony9402/baekjoon> 

- 김민상 (tony9402@naver.com)
 - (현) 송실대학교 대학원 3학기
 - (현) 각종 알고리즘 대회 검수 및 출제
 - (현) 취미로 알고리즘 푸는 중
 - 송실대학교 전자공학과 졸업
 - 송실대학교 컴퓨터공학과 (복수전공) 졸업
 - Software Mastro 11th 수료생

등수	37
맞은 문제	3503
맞았지만 만점을 받지 못한 문제	7
시도했지만 맞지 못한 문제	97
제출	13972
문제집	28
만든 문제	44
오타를 찾음	8
데이터를 추가	6
어색한 표현을 찾음	1
문제를 검수	394

2021-2022

2021
S
M
T
W
T
F
S 

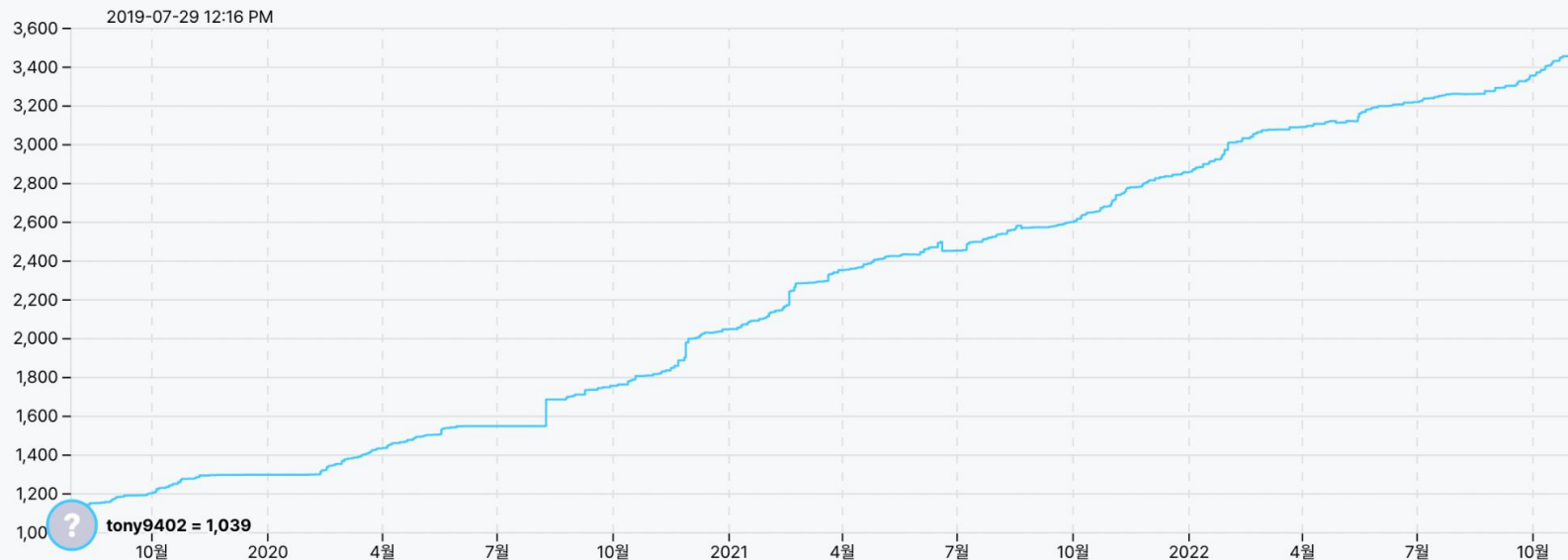
2022
S
M
T
W
T
F
S 

맞은 문제

발표자 소개

문 문제 수

전체 1년 6개월 3개월 1개월 2주 1주 1일



발표에 앞서...

주관적인 의견이 많습니다.

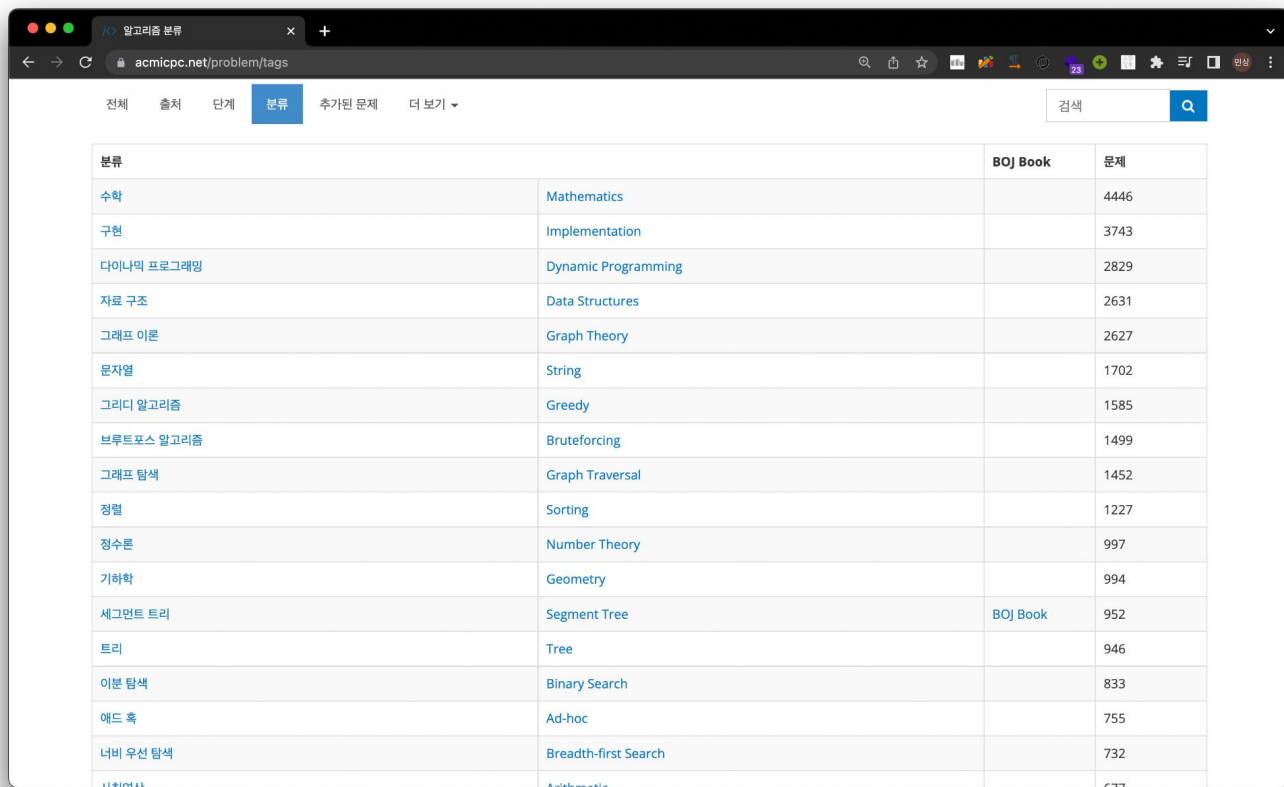
코딩테스트 준비에는 여러 가지 길이 존재하는 한 가지 길만 있는 것이 아닙니다.

참고 정도만 하시면 좋을 것 같습니다.

알고리즘 종류

- BFS
- DFS
- Dijkstra
- Two Pointer
- Segment Tree
- Segment Tree with Lazy
- Aho-corasick
- Suffix Array
- Heavy-Light Decomposition
- Prefix Sum
- Sweeping
- Dynamic Programming
- Dynamic Programming on Tree
- Network Flow
- Minimum Spanning Tree
- ...

알고리즘 종류 (Baekjoon Online Judge)



분류	BOJ Book	문제
수학	Mathematics	4446
구현	Implementation	3743
다이나믹 프로그래밍	Dynamic Programming	2829
자료 구조	Data Structures	2631
그래프 이론	Graph Theory	2627
문자열	String	1702
그리디 알고리즘	Greedy	1585
브루트포스 알고리즘	Bruteforcing	1499
그래프 탐색	Graph Traversal	1452
정렬	Sorting	1227
정수론	Number Theory	997
기하학	Geometry	994
세그먼트 트리	Segment Tree	BOJ Book 952
트리	Tree	946
이분 탐색	Binary Search	833
애드 혹	Ad-hoc	755
너비 우선 탐색	Breadth-first Search	732
산술연산	Arithmetic	677

총 194가지 알고리즘이 존재

기업 코딩테스트에는 저런 알고리즘이 다 나올까?

절대 아니다.

그럼 기업 코딩테스트에 어떤 유형들이 나올까?

- 자료구조 (큐, 스택, 힙 등)
- 구현
- 시뮬레이션
- 완전탐색
- 백트래킹
- 문자열
- 수학 (소수 판별, GCD, LCM)
- 그리디
- 다이나믹 프로그래밍
- 그래프탐색 (BFS, DFS)
- 분할정복
- 누적 합
- 트리
- 두 포인터
- 위상정렬
- 최단거리 (다익스트라, BFS, 플로이드)
- 최소 스패닝 트리

과연 그럴까요?

최근 기업 코딩테스트에 어떤 유형들이 나올까?

삼성 - SW 역량테스트 2022 하반기

48	<u>싸움땅</u>	삼성 SW 역량테스트 2022 하반기 오전 1번 문제	보통
49	<u>산타의 선물 공장</u>	삼성 SW 역량테스트 2022 하반기 오전 2번 문제	어려움
50	<u>코드트리 빵</u>	삼성 SW 역량테스트 2022 하반기 오후 1번 문제	보통
51	<u>산타의 선물 공장 2</u>	삼성 SW 역량테스트 2022 하반기 오후 2번 문제	어려움

- BFS, DFS
- 시뮬레이션
- 자료구조

최근 기업 코딩테스트에 어떤 유형들이 나올까?

삼성 - SW 역량테스트 2022 상반기

44	<u>술래잡기</u>	삼성 SW 역량테스트 2022 상반기 오전 1번 문제	어려움
45	<u>예술성</u>	삼성 SW 역량테스트 2022 상반기 오전 2번 문제	보통
46	<u>꼬리잡기놀이</u>	삼성 SW 역량테스트 2022 상반기 오후 1번 문제	어려움
47	<u>나무박멸</u>	삼성 SW 역량테스트 2022 상반기 오후 2번 문제	보통

- BFS, DFS
- 시뮬레이션
- 자료구조

최근 기업 코딩테스트에 어떤 유형들이 나올까?

카카오 - 2022 테크 여름인턴십

행렬과 연산 2022 KAKAO TECH INTERNSHIP	Lv. 4	256명	8%
등산코스 정하기 2022 KAKAO TECH INTERNSHIP	Lv. 3	889명	22%
코딩 테스트 공부 2022 KAKAO TECH INTERNSHIP	Lv. 3	752명	17%
두 큐 합 같게 만들기 2022 KAKAO TECH INTERNSHIP	Lv. 2	3,803명	43%
성격 유형 검사하기 2022 KAKAO TECH INTERNSHIP	Lv. 1	6,709명	40%

- 구현
- 그리디
- 다이나믹 프로그래밍
- 다익스트라
- 자료구조 (덱, 리스트)

최근 기업 코딩테스트에 어떤 유형들이 나올까?

카카오 - 2022 카카오 신입 공채

사라지는 발판 2022 KAKAO BLIND RECRUITMENT	Lv. 3	673명	11%
파괴되지 않은 건물 2022 KAKAO BLIND RECRUITMENT	Lv. 3	2,217명	35%
양과 늑대 2022 KAKAO BLIND RECRUITMENT	Lv. 3	2,174명	32%
양궁대회 2022 KAKAO BLIND RECRUITMENT	Lv. 2	4,064명	32%
주차 요금 계산 2022 KAKAO BLIND RECRUITMENT	Lv. 2	7,403명	57%
k진수에서 소수 개수 구하기 2022 KAKAO BLIND RECRUITMENT	Lv. 2	6,933명	58%
신고 결과 받기 2022 KAKAO BLIND RECRUITMENT	Lv. 1	21,591명	32%

- 자료구조
- 수학 (소수 판별)
- 문자열 (파싱)
- 완전탐색
- 구현
- 누적 합

최근 기업 코딩테스트에 어떤 유형들이 나올까?

기타 회사

- **A사**

- 슬라이딩 윈도우
- 백트래킹
- 문자열 (파싱)
- **BFS**

- **B사**

- 구현
- 이분탐색
- 누적 합

- **C사**

- 그리디
- 정렬

- **D사**

- 문자열
- 트리
- 백트래킹
- 그래프 탐색 (**BFS, DFS**)
- 다이나믹 프로그래밍

최근 기업 코딩테스트에 어떤 유형들이 나올까?

- 자료구조 (큐, 스택, 힙 등)
- 구현
- 시뮬레이션
- 완전탐색
- 백트래킹
- 문자열
- 수학 (소수 판별, GCD, LCM)
- 그리디
- 다이나믹 프로그래밍
- 그래프탐색 (BFS, DFS)
- 분할정복
- 누적 합
- 트리
- 두 포인터
- 위상정렬
- 최단거리 (다익스트라, BFS, 플로이드)
- 최소 스패닝 트리

그래서 알고리즘 공부는 어떻게 할까?

0. 언어 선택 (Java, C/C++, Python, Swift 등)

1. 자주 나오는 알고리즘, 자료구조 공부

2. 시간제한을 두고 풀어보기

3. 꾸준히 풀어서 감 잃지 않기

그래서 알고리즘 공부는 어떻게 할까?

0. 언어 선택

- 가장 편한 언어
- 지원할 직군 관련 언어
- 서브 언어 (선택)

언어: Python, Java, C/C++, Swift, Javascript 등

그래서 알고리즘 공부는 어떻게 할까?

1. 자주 나오는 알고리즘, 자료구조 공부





- 기업 코딩테스트는 필수로 알아야 하는 알고리즘, 자료구조에서만 나온다.

<https://github.com/tony9402/baekjoon>

순번	Tag	태그	문제집	추천 문제 수	총 문제 수
00	Data Structure	자료구조	바로가기	16	25
01	Data Structure 2	자료구조 2	바로가기	11	21
02	Tree	트리	바로가기	15	44
03	Math	수학	바로가기	18	29
04	Greedy	탐욕법	바로가기	27	62
05	Dynamic Programming 1	동적계획법 1	바로가기	27	80
06	Dynamic Programming 2	동적계획법 2	바로가기	28	81
07	Two Pointer	투 포인터	바로가기	13	36
08	Implementation	구현	바로가기	30	51
09	Graph Traversal	그래프 탐색	바로가기	31	96
10	Brute Force	완전탐색	바로가기	33	94
11	Simulation	시뮬레이션	바로가기	30	48
12	Binary Search	이분탐색	바로가기	20	66
13	Backtracking	백트래킹	바로가기	28	75
14	Divide and conquer	분할정복	바로가기	08	18
15	Prefix Sum	누적 합	바로가기	11	32
16	String	문자열	바로가기	19	63
17	Shortest Path	최단거리	바로가기	18	53
18	Topological Sorting	위상정렬	바로가기	05	14
19	Disjoint Set	분리 집합	바로가기	06	20
20	Minimum Spanning Tree(MST)	최소 스패닝 트리	바로가기	08	24

















그래서 알고리즘 공부는 어떻게 할까?

1. 자주 나오는 알고리즘, 자료구조 공부

- 기업 코딩테스트는 필수로 알아야하는 알고리즘, 자료구조에서만 나온다.  
- solved.ac 는 (브론즈 5) 부터 (루비 1)까지 문제의 난이도를 붙이는 프로젝트 (론즈, 실버, 골드, 플레티넘, 다이아, 루비)
- 기업 코딩테스트는 (브론즈 5) 부터 (플레티넘 5) 정도의 난이도로 출제 (주로 브론즈에서 골드 난이도로 출제)

그래서 알고리즘 공부는 어떻게 할까?

1. 자주 나오는 알고리즘, 자료구조 공부

5 Bronze V	0	99	99	
4 Bronze IV	15	151	166	
3 Bronze III	185	331	516	
2 Bronze II	452	351	803	
1 Bronze I	430	207	637	
5 Silver V	479	181	660	
4 Silver IV	519	177	696	
3 Silver III	526	166	692	
2 Silver II	567	178	745	
1 Silver I	540	178	718	
5 Gold V	688	210	898	
4 Gold IV	738	243	981	
3 Gold III	772	176	948	
2 Gold II	712	160	872	
1 Gold I	694	113	807	
5 Platinum V	708	112	820	

그래서 알고리즘 공부는 어떻게 할까?

1. 자주 나오는 알고리즘, 자료구조 공부

- 대회 유형 문제 등 다양하게 포함되어 있음

그래서 알고리즘 공부는 어떻게 할까?

1. 자주 나오는 알고리즘, 자료구조 공부

순번	추천 문제	문제 번호	문제 이름	난이도	풀이 링크
00	✓	1620	나는야 포켓몬 마스터 이다솜	4	바로가기
01	✓	14425	문자열 집합	3	바로가기
02	✓	11279	최대 힙	2	바로가기
03	✓	2075	N번째 큰 수	2	바로가기
04	✓	4358	생태학	2	바로가기
05	✓	11286	절댓값 힙	1	바로가기
06	✓	7662	이중 우선순위 큐	4	바로가기
07	✓	21939	문제 추천 시스템 Version 1	4	바로가기
08	✓	2696	중앙값 구하기	2	바로가기
09	✓	21942	부품 대여장	2	바로가기
10	✓	21944	문제 추천 시스템 Version 2	2	

그래서 알고리즘 공부는 어떻게 할까?

1. 자주 나오는 알고리즘, 자료구조 공부

- 가장 좋았던 방법은 여러 유형을 풀어보는 것이 좋음
- 전형적인 알고리즘 문제를 시작으로 응용 문제까지 풀어보는 것이 중요
- 많은 문제, 많은 유형을 경험해보면 코딩테스트 때 “어 이거 어디서 본 문젠데?” 느낌을 받음
- 구현 문제 꾸준히 풀어보기

그래서 알고리즘 공부는 어떻게 할까?

2. 시간제한을 두고 풀어보기

- 기출 문제를 풀 때 같은 제한 시간을 두고 풀어보기

그래서 알고리즘 공부는 어떻게 할까?

3. 꾸준히 풀어서 감 잃지 않기

- 한번 공부했다고 오랫동안 기억에 남는 것이 아니라 안 쓰다 보면 반드시 까먹게 됨.
- 1일 1문제, 1주일 3문제 등과 같이 꾸준히 문제를 푸는 것이 중요.

Tip

- 백준, 구름, 코드트리 와 같은 사이트는 입력부터 직접 짜야 하는 스타일
- 프로그래머스, **leetcode**는 함수 인자로 입력이 들어옴
- 자신이 볼 코딩테스트는 어떤 방식인지 확인 후 해당 환경에서 문제를 풀어봐야 함.

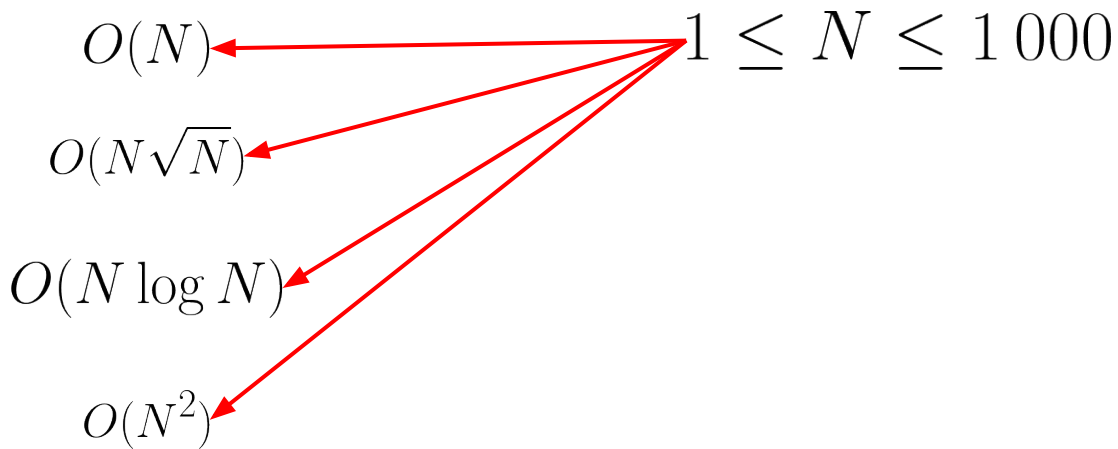
Tip

- 너무 쉬운 문제를 많이 푸는 것은 그렇게 중요하지 않음.
- 여러 코딩테스트를 한번 경험해보기 (서류 없이 볼 수 있는 공채나 모의 코딩테스트)
- 발전이 없는 것 같다고 해서 포기하지 말기.

Tip

코딩테스트 - 시간복잡도 계산을 하여 시간안에 통과되는지 계산할 줄
알아야 한다.

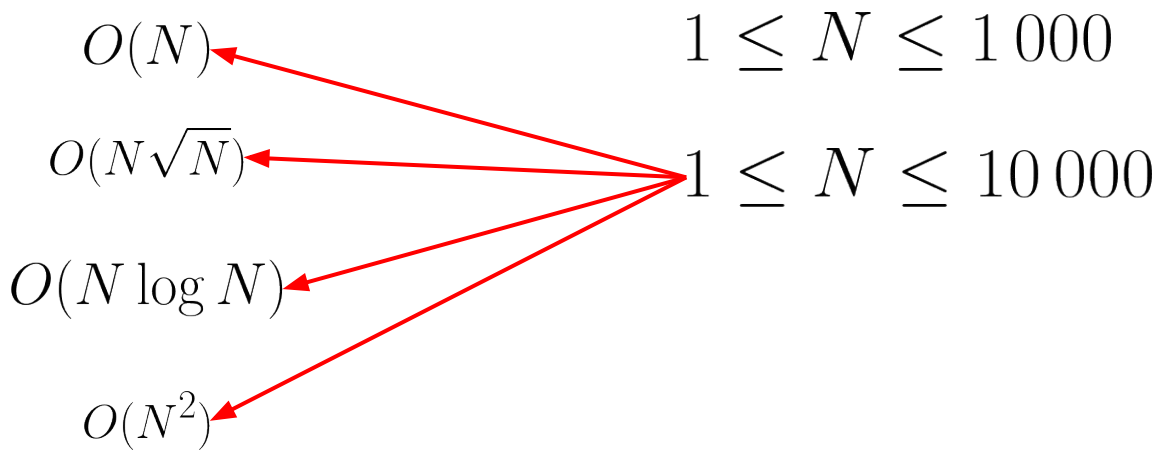
- 시간복잡도 : C++ 기준 1초에 약 1억번 연산을 진행한다고 가정



Tip

코딩테스트 - 시간복잡도 계산을 하여 시간안에 통과되는지 계산할 줄
알아야 한다.

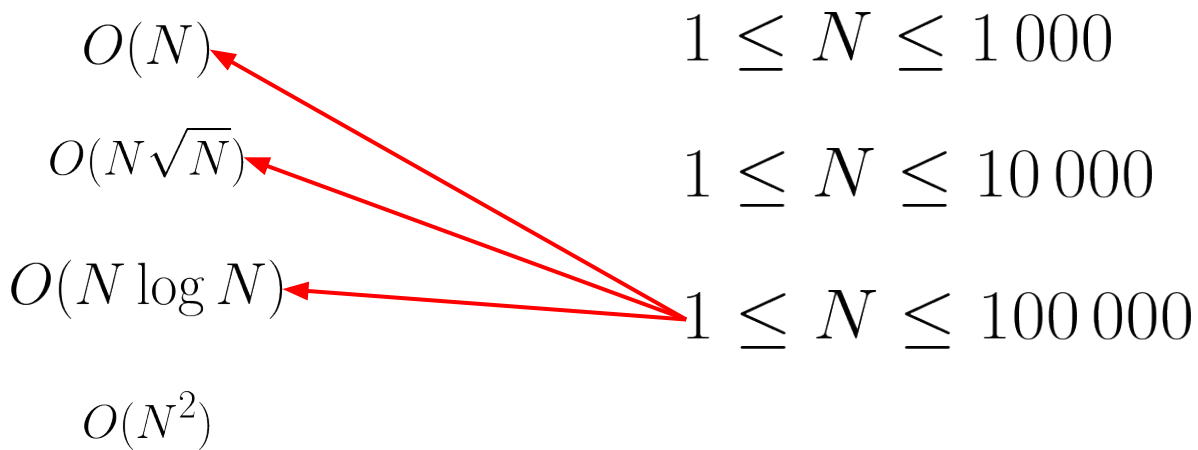
- 시간복잡도 : C++ 기준 1초에 약 1억번 연산을 진행한다고 가정



Tip

코딩테스트 - 시간복잡도 계산을 하여 시간안에 통과되는지 계산할 줄
알아야 한다.

- 시간복잡도 : C++ 기준 1초에 약 1억번 연산을 진행한다고 가정



Tip

코딩테스트 - 시간복잡도 계산을 하여 시간안에 통과되는지 계산할 줄
알아야 한다.

- 시간복잡도 : C++ 기준 1초에 약 1억번 연산을 진행한다고 가정

$O(N)$	$1 \leq N \leq 1\,000$
$O(N\sqrt{N})$	$1 \leq N \leq 10\,000$
$O(N \log N)$	$1 \leq N \leq 100\,000$
$O(N^2)$	$1 \leq N \leq 1\,000\,000$

Tip

지문에 답 또는 힌트가 숨어져 있을 수 있다.

상근이는 환경에 매우 관심이 많기 때문에, 나무를 필요한 만큼만 집으로 가져가려고 한다. 이때, 적어도 M미터의 나무를 집에 가져가기 위해서 절단기에 설정할 수 있는 높이의 최댓값을 구하는 프로그램을 작성하시오.

입력

첫째 줄에 나무의 수 N과 상근이가 집으로 가져가려고 하는 나무의 길이 M이 주어진다. ($1 \leq N \leq 1,000,000$, $1 \leq M \leq 2,000,000,000$)

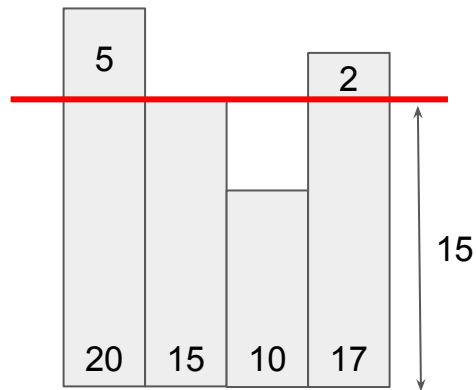
둘째 줄에는 나무의 높이가 주어진다. 나무의 높이의 합은 항상 M보다 크거나 같기 때문에, 상근이는 집에 필요한 나무를 항상 가져갈 수 있다. 높이는 1,000,000,000보다 작거나 같은 양의 정수 또는 0이다.

Tip

지문에 답 또는 힌트가 숨어져 있을 수 있다.

예제 입력 1 복사

```
4 7
20 15 10 17
```



Tip

지문에 답 또는 힌트가 숨어져 있을 수 있다.

상근이는 환경에 매우 관심이 많기 때문에, 나무를 필요한 만큼만 집으로 가져가려고 한다. 이때, 적어도 M미터의 나무를 집에 가져가기 위해서 절단기에 설정할 수 있는 높이의 최댓값을 구하는 프로그램을 작성하시오.

입력

첫째 줄에 나무의 수 N과 상근이가 집으로 가져가려고 하는 나무의 길이 M이 주어진다. ($1 \leq N \leq 1,000,000$, $1 \leq M \leq 2,000,000,000$)

둘째 줄에는 나무의 높이가 주어진다. 나무의 높이의 합은 항상 M보다 크거나 같기 때문에, 상근이는 집에 필요한 나무를 항상 가져갈 수 있다. 높이는 1,000,000,000보다 작거나 같은 양의 정수 또는 0이다.

O(H) : 시간
초과

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

Tip

지문에 답 또는 힌트가 숨어져 있을 수 있다.

$N \times M$ 크기의 배열로 표현되는 미로가 있다.

1	0	1	1	1	1
1	0	1	0	1	0
1	0	1	0	1	1
1	1	1	0	1	1

미로에서 1은 이동할 수 있는 칸을 나타내고, 0은 이동할 수 없는 칸을 나타낸다. 이러한 미로가 주어졌을 때, (1, 1)에서 출발하여 (N, M)의 위치로 이동할 때 지나야 하는 최소의 칸 수를 구하는 프로그램을 작성하시오. 한 칸에서 다른 칸으로 이동할 때, 서로 인접한 칸으로만 이동할 수 있다.

위의 예에서는 15칸을 지나야 (N, M)의 위치로 이동할 수 있다. 칸을 셀 때에는 시작 위치와 도착 위치도 포함한다.

입력

첫째 줄에 두 정수 N, M ($2 \leq N, M \leq 100$)이 주어진다. 다음 N 개의 줄에는 M 개의 정수로 미로가 주어진다. 각각의 수들은 붙어서 입력으로 주어진다.

출력

첫째 줄에 지나야 하는 최소의 칸 수를 출력한다. 항상 도착위치로 이동할 수 있는 경우만 입력으로 주어진다.

최단거리 알고리즘

- **BFS**
- 다익스트라

Tip

지문에 답 또는 힌트가 숨어져 있을 수 있다.

문제

방향그래프가 주어지면 주어진 시작점에서 다른 모든 정점으로의 최단 경로를 구하는 프로그램을 작성하시오. 단, 모든 간선의 가중치는 10 이하의 자연수이다.

입력

첫째 줄에 정점의 개수 V 와 간선의 개수 E 가 주어진다. ($1 \leq V \leq 20,000$, $1 \leq E \leq 300,000$) 모든 정점에는 1부터 V 까지 번호가 매겨져 있다고 가정한다. 둘째 줄에는 시작 정점의 번호 K ($1 \leq K \leq V$)가 주어진다. 셋째 줄부터 E 개의 줄에 걸쳐 각 간선을 나타내는 세 개의 정수 (u, v, w)가 순서대로 주어진다. 이는 u 에서 v 로 가는 가중치 w 인 간선이 존재한다는 뜻이다. u 와 v 는 서로 다르며 w 는 10 이하의 자연수이다. 서로 다른 두 정점 사이에 여러 개의 간선이 존재할 수도 있음에 유의한다.

출력

첫째 줄부터 V 개의 줄에 걸쳐, i 번째 줄에 i 번 정점에서의 최단 경로의 경로값을 출력한다. 시작점 자신은 0으로 출력하고, 경로가 존재하지 않는 경우에는 INF를 출력하면 된다.

최단거리 알고리즘

- BFS
- 다익스트라

Tip

수의 범위를 기억하자.

- int(Java: int)는 $\sim 2^{31}$ 부터 $2^{31}-1$ 까지 수를 표현할 수 있다.
- long long(Java: long)은 $\sim 2^{63}$ 부터 $2^{63}-1$ 까지 수를 표현할 수 있다.

```
int L = 0, R = 2000000000;  
while(L <= R) {  
    int mid = (L + R) / 2;  
    if(OK(mid)) L = mid + 1;  
    else R = mid - 1;  
}
```

```
long long L = 0, R = 2000000000;  
while(L <= R) {  
    long long mid = (L + R) / 2;  
    if(OK(mid)) L = mid + 1;  
    else R = mid - 1;  
}
```

Tip

수의 범위를 기억하자.

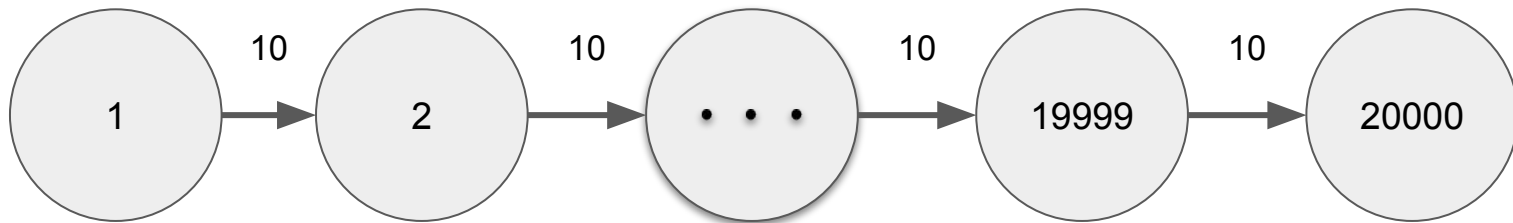
방향그래프가 주어지면 주어진 시작점에서 다른 모든 정점으로의 최단 경로를 구하는 프로그램을 작성하시오. 단, 모든 간선의 가중치는 10 이하의 자연수이다.

입력

첫째 줄에 정점의 개수 V 와 간선의 개수 E 가 주어진다. ($1 \leq V \leq 20,000$, $1 \leq E \leq 300,000$) 모든 정점에는 1부터 V 까지 번호가 매겨져 있다고 가정한다. 둘째 줄에는 시작 정점의 번호 K ($1 \leq K \leq V$)가 주어진다. 셋째 줄부터 E 개의 줄에 걸쳐 각 간선을 나타내는 세 개의 정수 (u, v, w) 가 순서대로 주어진다. 이는 u 에서 v 로 가는 가중치 w 인 간선이 존재한다는 뜻이다. u 와 v 는 서로 다르며 w 는 10 이하의 자연수이다. 서로 다른 두 정점 사이에 여러 개의 간선이 존재할 수도 있음에 유의한다.

Tip

수의 범위를 기억하자.

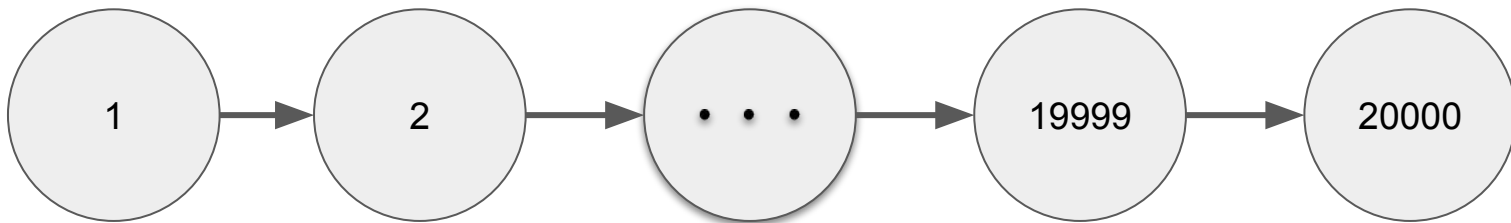


가중치 * 간선의 개수 = $10 * (20,000 - 1) = 199,990 \leq \text{INT_MAX}$

Tip

수의 범위를 기억하자.

가중치가 최대 200,000라면?



가중치 * 간선의 개수 = $200,000 * (20,000 - 1) = \text{약 } 40\text{억} > \text{INT_MAX}$

Tip

구현 또는 시뮬레이션 문제는 모듈화를 하자.

1. 공장 설립
2. 물건 모두 옮기기
3. 앞 물건만 교체하기
4. 물건 나누기
5. 선물 정보 얻기
6. 벨트 정보 얻기

```
int main() {
    ios::sync_with_stdio(false);
    cin.tie(0);
    int Q; cin >> Q;
    while(~--Q) {
        int cmd; cin >> cmd;
        if(cmd == 100) Build();
        else if(cmd == 200) cout<<Move()<<'\n';
        else if(cmd == 300) cout<<Swap()<<'\n';
        else if(cmd == 400) cout<<Divide()<<'\n';
        else if(cmd == 500) cout<<GiftInfo()<<'\n';
        else if(cmd == 600) cout<<BeltInfo()<<'\n';
    }
    return 0;
}
```

<https://www.codetree.ai/frequent-problems/santa-gift-factory-2/description>

공부자료 소개

- [tony9402 : 알고리즘 별 추천문제](#)
- [BaaaaaaaaaarkingDog : 실전 알고리즘](#)
- [류호석 : 코딩테스트 알고리즘](#)
- [나동빈 : 이것이 취업을 위한 코딩 테스트다](#)
- [백준 온라인 저지](#)
- [프로그래머스](#)
- [코드트리](#)
- [leetcode](#)
- [SW Expert Academy](#)
- ...

https://github.com/tony9402/baekjoon/blob/main/link_for_study.md

공부자료 소개

코드트리 - 진단 테스트

학습 진단을 위한 테스트 안내

코딩테스트 준비를 위한 맞춤 학습 프로그램을 제공하기 위해
간단한 코딩 테스트를 진행해볼거예요.

이전

다음

<https://www.codetree.ai/landing/test-guide/1>

Q&A

Q&A

- 문제를 풀 때 정답을 언제 봐야할까?

머칠이 걸리더라도 최대한 고민하다가 보는 것이
좋다.

몇 분에서 몇 시간 동안 고민해도 모르겠으면 바로
본다.

둘 다 틀린 것이 아니다.

Q&A

- 다른 사람의 코드를 보는 것이 좋을까요?

보는 것이 좋다.

```
int dfs(int y, int x) {
    Map[y][x] = '0';
    int ret = 1;
    if(y + 1 < N && Map[y + 1][x] == '1') ret += dfs(y + 1, x);
    if(y - 1 >= 0 && Map[y - 1][x] == '1') ret += dfs(y - 1, x);
    if(x + 1 < N && Map[y][x + 1] == '1') ret += dfs(y, x + 1);
    if(x - 1 >= 0 && Map[y][x - 1] == '1') ret += dfs(y, x - 1);
    return ret;
}
```

```
const int dy[] = {-1,1,0,0};
const int dx[] = {0,0,-1,1};

int dfs(int y, int x) {
    Map[y][x] = '0';
    int ret = 1;
    for(int k = 0; k < 4; ++k) {
        int nexty = y + dy[k], nextx = x + dx[k];
        if(0 > nexty || nexty >= N || 0 > nextx || nextx >= N) continue;
        if(Map[nexty][nextx] == '0') continue;
        ret += dfs(nexty, nextx);
    }
    return ret;
}
```


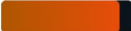

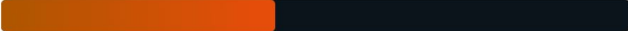
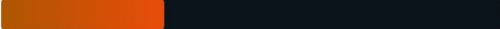
Q&A

- 다른 사람의 코드를 보는 것이 좋을까요?

🔖 1753번	제출	맞힌 사람	숫코딩	재채점 결과	채점 현황	내 제출	🔗 난이도 기여	강의 ▾
질문 검색								
모든 언어								
모든 언어								
언어 그룹								
C++								
Java								
Python								
C								
Rust								
언어								
C++17								
C++v								
틀렸습니다								
시간 초과								
메모리 초과								
출력 초과								
런타임 에러								
컴파일 에러								

Q&A

- 알고리즘 풀다가 번아웃이 왔을 때 어떻게 대처하셨나요?

5 Bronze V	0	99	99	
4 Bronze IV	17	151	168	
3 Bronze III	185	332	517	
2 Bronze II	453	349	802	
1 Bronze I	430	208	638	

참고하면 좋을 영상

- 코딩 테스트 공부, 나는 잘 하고 있는걸까? - 류호석
- 처음 시작하는 코딩테스트 준비법 - 썸튜브
- 네카라 합격자의 코딩테스트 준비 비결 - 한끗차IT

감사합니다