



신촌지역 대학생 프로그래밍 동아리 연합 여름 대회

SUAPC 2026 Summer

Main Contest Problemset

본대회 문제지

2026년 8월 25일 11:30 - 16:30
dcamp 마포 5층 박병원홀 (마포대로 122)

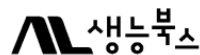
주최

ICPC Sinchon

후원



Dify



dcamp

cenix820 cubic jtw7913 nanoka ramma solved.ac

김태우 김태윤(ystaeyoon113) 안녕원팬 이재열 임예찬

문제 목록

문제지는 양면 인쇄되었으며, 30페이지 15장 14문제입니다. 문제지의 장수와 인쇄 상태를 잘 확인하시고, 문제지에 오류가 있는 것 같다면 즉시 손을 들어 주십시오.

각 문제에 대한 정보는 다음과 같습니다:

#	TL	ML	제목
A	1 초	1024 MB	수앞C
B	1 초	1024 MB	분탕의 신 안녕원
C	1 초	1024 MB	공 넣기
D	1 초	1024 MB	Regulating Bracket Sequence
E	1 초	1024 MB	K분 그래프 3
F	1 초	1024 MB	122333
G	3 초	1024 MB	쌀 가져가기와 쿼리
H	2 초	1024 MB	README
I	2 초	1024 MB	책 정렬
J	2 초	1024 MB	트리 뽑아내기
K	1 초	1024 MB	술 푸는 사람
L	1 초	1024 MB	정투상도
M	4 초	1024 MB	쌀 재배하기와 쿼리
N	1 초	1024 MB	매직 쌀 2

문제를 잘 읽는 것이 Dify 해커톤에 도움이 될 수도 있습니다.

그럼, Good Luck and Have Fun!

문제 A. 수앞C

시간 제한 1 초 메모리 제한 1024 MB

길이 N 의 영문 대문자와 숫자로만 이루어진 문자열이 주어진다. 문자열에 포함된 **C** 중 숫자 바로 앞에 있는 것의 개수를 구해 보자.

입력

첫 번째 줄에 문자열의 길이 N 이 주어진다. ($1 \leq N \leq 100$)

두 번째 줄에 길이 N 의 문자열이 주어진다.

출력

문자열에 포함된 **C** 중 숫자 바로 앞에 있는 것의 개수를 출력하여라.

입출력 예시

표준 입력(stdin)		표준 출력(stdout)
10	SUAPC2026S	1
6	C4ESAR	1

이 페이지는 공백입니다

문제 B. 분탕의 신 안녕원

시간 제한 1 초 메모리 제한 1024 MB

이 문제는 인터랙티브 투 스텝 문제이다.

다다스와 루날리티는 하나의 트리를 이용해 정수 X 를 전달하려 한다. 다다스와 루날리티는 X 를 전달받기 전에 사전에 서로 합의하여 N 개의 정점으로 이루어진 트리 하나를 미리 정해 둘 수 있다. 이후 다다스가 루날리티에게 전달해야 할 정수 X 를 알게 되면, 다다스는 트리의 모든 정점을 검은색 또는 흰색으로 칠한다.

그러나 이들의 계획을 알게 된 안녕원은 트리가 루날리티에게 무사히 전달되는 모습을 보고 싶지 않았다. 안녕원은 임의의 정점 u 를 선택한 뒤, u 자신과 u 에 인접한 모든 정점의 색을 반전하는 연산을 **원하는 만큼** 수행할 수 있다. 색을 반전한다는 것은 검은색 정점을 흰색으로, 흰색 정점을 검은색으로 바꾸는 것을 뜻한다. 안녕원은 충분히 분탕을 친 뒤 모든 정점의 번호를 임의로 다시 매겨 트리를 루날리티에게 전달한다.

다다스와 루날리티는 정수 X 가 주어지기 전에 어떤 방식으로 행동할지 미리 약속해 둘 수 있다. 하지만 X 가 주어진 뒤 다다스가 루날리티에게 정보를 전달할 수 있는 방법은 트리의 정점을 색칠하는 것뿐이다. 루날리티는 안녕원의 방해가 모두 끝난 뒤 전달받은 트리의 구조와 각 정점의 색만을 보고, 다다스가 전달하려 했던 정수 X 를 정확히 알아내야 한다.

안녕원이 어떤 방식으로 트리를 뒤섞더라도 루날리티가 항상 X 를 알아낼 수 있도록 다다스와 루날리티의 전략을 직접 구현해 보자.

이 문제에서 프로그램은 하나의 테스트 데이터에 대해 두 번 실행된다. 첫 번째 실행에서는 다다스의 역할을 수행하고, 두 번째 실행에서는 루날리티의 역할을 수행해야 한다.

인터랙션

첫 번째 줄에 이번 실행에서 맡은 역할을 나타내는 문자열 R 이 주어진다. R 은 **dadas08** 또는 **lunarility** 중 하나이다. **dadas08**은 이번 실행에서 다다스의 역할을 수행해야 함을, **lunarility**는 루날리티의 역할을 수행해야 함을 의미한다.

First Step (**dadas08**)

양의 정수 N 을 출력하여야 한다. ($1 \leq N \leq 160$)

그다음 줄부터 $N-1$ 개의 줄에 걸쳐 그중 i 번째 줄에 두 정수 u_i, v_i 를 공백으로 구분하여 출력하여야 한다. 이는 정점 u_i 와 정점 v_i 를 잇는 간선이 존재함을 의미한다. 출력한 $N-1$ 개의 간선은 다다스와 루날리티가 사전에 정한 N 개의 정점으로 이루어진 트리여야 한다. 모든 i 에 대해 $1 \leq u_i, v_i \leq N$ 이어야 한다.

그다음 줄에 테스트 케이스의 수를 나타내는 양의 정수 T 가 주어진다. ($1 \leq T \leq 1000$)

각 테스트 케이스는 다음과 같은 순서로 진행된다.

- 루날리티에게 전달해야 할 정수 X 가 주어진다. ($1 \leq X \leq 10^9$)
- X 를 입력받은 직후, 길이가 N 인 문자열 하나를 출력하여야 한다. 출력하는 문자열은 **W**와 **B**로만 이루어져야 하며, 문자열의 i 번째 문자는 i 번 정점의 색을 나타낸다. **W**는 흰색을, **B**는 검은색을 의미한다.

이 과정을 총 T 번 반복한다. 즉, 정수 X 하나를 입력받을 때마다 그에 대응하는 문자열 하나를 출력하여야 한다.

Second Step (**lunarility**)

그다음 줄에 양의 정수 N 이 주어진다. ($1 \leq N \leq 160$)

그다음 $N-1$ 개의 줄에 두 정수 u_i, v_i 가 공백으로 구분되어 주어진다. 이는 정점 u_i 와 정점 v_i 를 잇는 간선이 존재함을 의미한다. 주어진 $N-1$ 개의 간선은 다다스와 루날리티가 사전에 정한 트리의 정점 번호를 임의로 다시 매긴 결과이다. 모든 i 에 대해 $1 \leq u_i, v_i \leq N$ 이다.

그다음 줄에 테스트 케이스의 수를 나타내는 양의 정수 T 가 주어진다. ($1 \leq T \leq 1000$)

각 테스트 케이스는 다음과 같은 순서로 진행된다.

- 길이가 N 인 문자열 하나가 주어진다. 문자열은 **W**와 **B**로만 이루어져 있으며, 문자열의 i 번째 문자는 i 번 정점의 색을 나타낸다. 이 문자열은 다다스가 정점들을 색칠한 뒤 안녕원이 원하는 만큼 연산을 수행하고, 정점의 번호를 임의로 다시 매긴 결과이다.
- 문자열을 입력받은 직후, 다다스가 전달하려 했던 정수 X 를 한 줄에 출력하여야 한다.

이 과정을 총 T 번 반복한다. 즉, 문자열 하나를 입력받을 때마다 그에 대응하는 정수 X 하나를 출력하여라.

출력 후에는 표준 출력 버퍼를 flush해 주어야 한다. 표준 출력 버퍼를 flush하지 않으면 채점 프로그램이 출력을 전달받지 못해 프로그램이 계속 대기할 수 있으며, 이 경우 오답 판정을 받을 수 있다.

인터랙터는 **적응적**이다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
dadas08	5
	1 2
	2 3
	3 4
	4 5
2	
7	BWBWW
11	WWBBB
lunarlity	
5	
2 1	
3 5	
1 4	
5 2	
2	
WBBWW	
	7
WWWBB	
	11

노트

예시의 빈 줄은 가독성을 위한 것이며, 실제로 주어지지 않는다.

문제 C. 공 넣기

시간 제한 1 초 메모리 제한 1024 MB

N 개의 상자가 일렬로 놓여 있다. 처음에는 모든 상자가 비어 있다.

한 번의 시행에서는 서로 이웃한 두 상자가 함께 선택되지 않도록 상자의 집합 S 를 고른다. 그 뒤 S 에 속한 각 상자에 공을 1개씩 넣는다.

모든 시행을 마친 뒤 i 번째 상자에 들어 있는 공의 개수가 a_i 가 되도록 하려고 한다. 필요한 최소 시행 횟수를 구하고, 그 방법을 출력하여라.

입력

첫 번째 줄에 상자의 개수 N 이 주어진다. ($1 \leq N \leq 200\,000$)

두 번째 줄에 N 개의 정수 $a_1, a_2, \dots, a_N$ 이 공백으로 구분되어 주어진다.

($0 \leq a_1, a_2, \dots, a_N \leq 200\,000$, $a_1 + a_2 + \dots + a_N \leq 200\,000$)

출력

첫 번째 줄에 필요한 최소 시행 횟수 K 를 출력하여라.

두 번째 줄부터 K 개의 줄에 각 시행의 정보를 출력하여라. 이 중 t 번째 줄에는 t 번째 시행에서 선택하는 상자의 개수 s_t 와, 선택하는 상자의 번호 $i_{t,1}, i_{t,2}, \dots, i_{t,s_t}$ 를 공백으로 구분하여 출력하여라.

상자 번호는 오름차순이어야 하며, 모든 $1 \leq j < s_t$ 에 대해 $i_{t,j+1} - i_{t,j} \geq 2$ 를 만족해야 한다.

출력한 시행을 모두 수행한 뒤, 모든 $1 \leq i \leq N$ 에 대해 i 번째 상자에 공이 정확히 a_i 개 들어 있어야 한다.

조건을 만족하는 방법이 여러 가지 존재한다면 그중 아무거나 출력한다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5	4
2 1 2 2 1	3 1 3 5
	2 1 3
	1 4
	2 2 4

이 페이지는 공백입니다

문제 D. Regulating Bracket Sequence

시간 제한 1 초 메모리 제한 1024 MB

다음 조건을 만족하는 문자열을 **균형 잡힌 괄호 문자열**이라고 한다.

- 빈 문자열은 균형 잡힌 괄호 문자열이다.
- S 가 균형 잡힌 괄호 문자열이라면 (S) , $\{S\}$, $[S]$ 도 각각 균형 잡힌 괄호 문자열이다.
- S 와 T 가 둘 다 균형 잡힌 괄호 문자열이라면 ST 도 균형 잡힌 괄호 문자열이다.
- 위 방식으로 만들 수 없는 문자열은 균형 잡힌 괄호 문자열이 아니다.

두 균형 잡힌 괄호 문자열 S 와 T 가 주어질 때, 다음 연산을 0회 이상 사용하여 S 를 T 로 만들 수 있는지 판별해보자.

- 두 균형 잡힌 괄호 문자열 A 와 B 에 대해 S 가 $(A)B$ 라면 S 를 $(B)A$ 또는 $B(A)$ 로 바꾼다.
- 두 균형 잡힌 괄호 문자열 A 와 B 에 대해 S 가 $\{A\}B$ 라면 S 를 $\{B\}A$ 또는 $B\{A\}$ 로 바꾼다.
- 두 균형 잡힌 괄호 문자열 A 와 B 에 대해 S 가 $[A]B$ 라면 S 를 $[B]A$ 또는 $B[A]$ 로 바꾼다.

입력

첫 번째 줄에 균형 잡힌 괄호 문자열 S 가 주어진다. ($2 \leq |S| \leq 500000$)

두 번째 줄에 균형 잡힌 괄호 문자열 T 가 주어진다. ($2 \leq |T| \leq 500000$)

출력

주어진 연산을 0회 이상 사용하여 S 를 T 로 만들 수 있다면 **YES**를, 아니면 **NO**를 출력하여라.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
{ } [] () [] () { }	YES
() [] { } { } [] ()	NO

첫 번째 예제의 경우 연산을 다음과 같이 적용하여 S 를 T 로 만들 수 있다.

- $\{ \}$ 에 연산을 적용하여 $\{ \}$ 로 만든다.
- $\{ \}$ 에 연산을 적용하여 $\{ \}$ 로 만든다.
- $\{ \}$ 에 연산을 적용하여 $\{ \}$ 로 만든다.

두 번째 예제의 경우 연산을 어떻게 적용해도 S 를 T 로 만들 수 없다.

이 페이지는 공백입니다

문제 E. K분 그래프 3

시간 제한 1 초 메모리 제한 1024 MB

정점이 N 개, 간선이 M 개인 가중치 유형 그래프가 주어진다. 정점은 1번부터 N 번까지 번호가 매겨져 있다.

각 간선은 어떤 정점 a_i 에서 정점 b_i 로 가는 방향을 가지며, 가중치 c_i 를 가진다.

이 그래프는 강하게 연결되어 있다. 즉, 임의의 두 정점 u, v 에 대해 u 에서 v 로 가는 유형 경로가 적어도 하나 존재한다.

정점 u 에서 정점 v 로 가는 워크(walk)란, 시작점이 u , 끝점이 v 인 정점과 간선의 열을 말한다. 워크에서는 같은 정점이나 같은 간선을 여러 번 방문해도 된다.

워크의 가중치는 그 워크에서 지난 모든 간선의 가중치의 합이다.

양의 정수 K 와 Q 개의 쿼리가 주어진다. 각 쿼리는 두 정수 u, v ($u \neq v$)로 이루어진다.

각 쿼리마다 다음 조건을 만족하는 정수 k 의 개수를 구하여야.

- $1 \leq k \leq K$
- 정점 u 에서 출발하여 정점 v 에서 끝나는 워크 중 가중치가 k 의 배수인 워크가 하나 이상 존재한다.

입력

첫 번째 줄에 네 개의 정수 N, M, Q, K 가 공백으로 구분되어 주어진다. ($2 \leq N, M, Q \leq 200000$; $1 \leq K \leq 10^9$)

그다음 줄부터 M 개의 줄 중 i 번째 줄에 세 정수 a_i, b_i, c_i 가 공백으로 구분되어 주어진다. 이는 a_i 에서 b_i 로 가는 가중치 c_i 의 유형 간선을 나타낸다. ($1 \leq a_i, b_i \leq N$; $a_i \neq b_i$; $0 \leq c_i \leq 10^9$)

그다음 줄부터 Q 개의 줄 중 j 번째 줄에 두 정수 u_j, v_j 가 공백으로 구분되어 주어진다. ($1 \leq u_j, v_j \leq N$; $u_j \neq v_j$)

주어지는 그래프는 강하게 연결된 그래프이다.

출력

각 쿼리에 대해 조건을 만족하는 정수 k 의 개수를 한 줄에 하나씩 출력하여야.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 3 3 10	3
1 2 1	5
2 3 2	7
3 1 3	
1 2	
1 3	
2 3	

첫 번째 예제의 그래프는 다음 유형 사이클이다. $1 \xrightarrow{1} 2 \xrightarrow{2} 3 \xrightarrow{3} 1$

사이클을 한 바퀴 돌 때마다 가중치 합이 6 증가한다.

예를 들어 질의 (1,3)에서 가능한 워크의 가중치 합은

$$3 + 6t \quad (t \geq 0)$$

꼴이다. 이 중 $1 \leq k \leq 10$ 에서 이를 만족하는 값은

1, 3, 5, 7, 9

이므로 답은 5이다.

이 페이지는 공백입니다

문제 F. 122333

시간 제한 1 초 메모리 제한 1024 MB

크기가 N 인 정수 배열 $A: a_1, a_2, \dots, a_N$ 이 주어진다. 당신은 배열의 원소를 선택하며 여러 단계를 순서대로 수행하려 한다.

i 번째 단계에서는 다음 조건을 모두 만족하도록 정확히 i 개의 원소를 선택해야 한다.

- 선택한 i 개 원소의 값은 모두 같아야 한다.
- $i \geq 2$ 라면, 선택한 모든 원소의 위치는 $(i-1)$ 번째 단계에서 선택한 모든 원소들의 위치보다 커야 한다.

아직 아무 단계도 수행하지 않았다면 위치에 대한 제한 없이 하나의 원소를 선택하여 첫 번째 단계를 수행할 수 있다. 수행할 수 있는 단계의 최대 횟수를 구하여라.

입력

첫 번째 줄에 정수 N 이 주어진다. ($1 \leq N \leq 1000000$)

두 번째 줄에 $a_1, a_2, \dots, a_N$ 이 공백으로 구분되어 주어진다. ($1 \leq a_i \leq N$)

출력

첫 번째 줄에 수행할 수 있는 단계의 최대 횟수를 출력하여라.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
9 2 1 4 1 3 5 3 4 3	3
10 4 4 4 4 3 3 3 2 2 1	3

이 페이지는 공백입니다

문제 G. 쌀 가져가기와 퀴리

시간 제한 3 초 메모리 제한 1024 MB

양의 정수 N, M 과 양의 정수로 이루어진 수열 $A: a_1, \dots, a_N, B: b_1, \dots, b_M, C: c_1, \dots, c_N, D: d_1, \dots, d_M$ 이 있다.

네 양의 정수 L_1, R_1, L_2, R_2 에 대해, $R_1 - L_1 + 1$ 명의 사람들과 $R_2 - L_2 + 1$ 종류의 쌀이 준비되어 있다. 모든 사람들은 협력하여 최대한 많은 쌀을 가져가려고 한다. 이때, 사람들이 쌀을 가져가기 위해서 다음 조건을 만족해야 한다.

- i 번째 사람은 쌀을 총 a_{L_1+i-1} 개까지만 가져갈 수 있다.
- j 번째 종류의 쌀은 총 b_{L_2+j-1} 개가 준비되어 있다.
- i 번째 사람이 j 번째 종류의 쌀을 최대 $c_{L_1+i-1} \times d_{L_2+j-1}$ 개만 가져갈 수 있다.

$f(L_1, R_1, L_2, R_2)$ 은 $R_1 - L_1 + 1$ 명의 사람들이 가져가는 쌀의 총합의 최댓값으로 정의한다.

당신에게 Q 개의 퀴리가 순서대로 주어진다. 각 퀴리는 5종류 중 하나로 주어지며, 종류는 다음과 같다.

- 1 $L_1 R_1 L_2 R_2$: $f(L_1, R_1, L_2, R_2)$ 의 값을 출력한다.
- 2 $i x$: a_i 를 x 로 바꾼다.
- 3 $j x$: b_j 를 x 로 바꾼다.
- 4 $i x$: c_i 를 x 로 바꾼다.
- 5 $j x$: d_j 를 x 로 바꾼다.

Q 개의 퀴리를 순서대로 처리해보자.

입력

첫 번째 줄에 N, M 이 공백으로 구분되어 주어진다. ($1 \leq N, M \leq 100000$)

두 번째 줄에 N 개의 정수 $a_1, a_2, \dots, a_N$ 이 공백으로 구분되어 주어진다. ($1 \leq a_i \leq 20$)

세 번째 줄에 M 개의 정수 $b_1, b_2, \dots, b_M$ 이 공백으로 구분되어 주어진다. ($1 \leq b_j \leq 20$)

네 번째 줄에 N 개의 정수 $c_1, c_2, \dots, c_N$ 이 공백으로 구분되어 주어진다. ($1 \leq c_i \leq 20$)

다섯 번째 줄에 M 개의 정수 $d_1, d_2, \dots, d_M$ 이 공백으로 구분되어 주어진다. ($1 \leq d_j \leq 20$)

여섯 번째 줄에 정수 Q 가 주어진다. ($1 \leq Q \leq 100000$)

이후 Q 개의 줄에 걸쳐 퀴리가 주어진다. ($1 \leq L_1 \leq R_1 \leq N; 1 \leq L_2 \leq R_2 \leq M; 1 \leq i \leq N; 1 \leq j \leq M; 1 \leq x \leq 20$)

1번 퀴리가 1번 이상 등장한다.

출력

Q 개의 줄에 걸쳐, 1번 퀴리의 결과를 순서대로 한 줄에 하나씩 출력하여야.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
2 3	3
1 2	6
1 2 3	
2 3	
2 3 4	
3	
1 1 2 1 3	
2 1 10	
1 1 2 1 3	

이 페이지는 공백입니다

문제 H. README

시간 제한 2 초 메모리 제한 1024 MB

읽어주셔서 감사합니다.

연결 무향 그래프 $G = (V, E)$ 가 주어진다.

정점 $v \in V$ 의 이심도 $e(v)$ 는 다음과 같이 정의한다.

$$e(v) = \max_{u \in V} \text{dist}(u, v).$$

그래프 G 의 반지름 $\text{rad}(G)$ 는 다음과 같이 정의한다.

$$\text{rad}(G) = \min_{v \in V} e(v).$$

G 의 스패닝 트리 T 중 $\text{rad}(T) = \text{rad}(G)$ 를 만족하는 것을 하나 찾아 출력하여라.

입력

첫 번째 줄에 그래프 G 의 정점 수 N 과 간선 수 M 이 공백으로 구분되어 주어진다. ($3 \leq N \leq 500$; $N-1 \leq M \leq 100000$)

그다음 줄부터 M 개의 줄 중 i 번째 줄에 두 정수 u_i, v_i 가 공백으로 구분되어 주어진다. 이는 i 번 간선이 정점 u_i 와 v_i 를 잇는다는 의미이다. ($1 \leq u_i, v_i \leq N$)

출력

조건을 만족하는 스패닝 트리를 구성하는 간선의 번호 $N-1$ 개를 공백으로 구분해 출력하여라.

정답이 여러 가지 존재한다면 그중 아무거나 출력한다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
8 11	2 8 10 1 11 3 5
5 8	
3 5	
4 7	
1 7	
4 6	
5 2	
6 7	
2 3	
6 8	
3 4	
1 2	

노트

$\text{dist}(u, v)$ 는 u 와 v 를 잇는 여러 단순 경로 중 경로를 구성하는 간선 개수의 최솟값이다.

그래프 $G = (V, E)$ 의 스패닝 트리란 정점 집합이 V 이고, 간선 집합이 E 의 부분집합이며, 트리인 그래프를 뜻한다. 스패닝 트리 T 의 반지름은 T 에서의 정점 간 거리를 이용해 같은 방식으로 정의한다.

이 페이지는 공백입니다

문제 I. 책 정렬

시간 제한 2 초 메모리 제한 1024 MB

책장에 1번부터 N 번까지 번호가 붙은 책이 한 권씩 있다. 초기에는 책장의 i 번째 위치에 a_i 번 책이 있다. 수아는 다음의 시행으로 책들을 정렬하여 i 번째 위치에 i 번 책이 놓이게 하려고 한다.

- 책 몇 권을 골라 그 순서를 유지한 채 맨 오른쪽으로 옮긴다. 예를 들어 책장에 13425 순으로 책이 있을 때 3번 책과 2번 책을 고르면 책의 순서가 14532가 된다.

수아는 위 시행의 횟수를 최소화하고 싶다. 시행을 최소로 하면서도 시행에서 옮겨야 하는 책들의 총 개수를 최소화하고 싶다. 시행횟수의 최솟값과 그때 옮겨야 하는 책의 총 개수의 최솟값을 구해보자.

입력

첫 번째 줄에 책의 수를 나타내는 정수 N 이 주어진다. ($1 \leq N \leq 5000$)

두 번째 줄에 서로 다른 N 개의 정수 $a_1, a_2, \dots, a_N$ 이 공백으로 구분되어 주어진다. ($1 \leq a_i \leq N$)

출력

시행 횟수의 최솟값과 그 때 옮겨야 하는 책의 총 개수의 최솟값을 공백으로 구분하여 출력하여야.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
4 2 1 4 3	2 3

이 페이지는 공백입니다

문제 J. 트리 뽑아내기

시간 제한 2 초 메모리 제한 1024 MB

정점이 N 개인 트리가 주어진다.

트리에서 정점이 하나 이상인 연결된 부분 그래프를 하나 선택한다. 선택한 부분 그래프 역시 트리이므로, 이를 **선택한 부분트리**라고 부르자. 선택한 부분트리 안에서 모든 정점의 차수가 K 이하여야 한다.

이후 선택한 부분트리에 포함된 모든 정점과, 그 정점들에 연결된 모든 간선을 제거한다. 남은 그래프는 여러 개의 연결 컴포넌트로 나뉠 수 있다. 남은 연결 컴포넌트들의 크기 중 최댓값을 가능한 한 작게 만들고자 한다. 모든 정점을 제거하여 연결 컴포넌트가 하나도 남지 않은 경우, 이 값은 0으로 정의한다.

$K = 1, 2, \dots, N$ 에 대해 가능한 최솟값을 A_K 라고 하자. $A_1, A_2, \dots, A_N$ 을 구하여라.

입력

첫 번째 줄에 트리의 정점 수 N 이 주어진다. ($1 \leq N \leq 300000$)

두 번째 줄부터 $N-1$ 개의 줄에 걸쳐 두 정수 u, v 가 공백으로 구분되어 주어진다. 이는 u 번 정점과 v 번 정점을 잇는 간선이 존재함을 의미한다. ($1 \leq u, v \leq N$)

출력

첫 번째 줄에 N 개의 정수 $A_1, A_2, \dots, A_N$ 을 공백으로 구분하여 출력하여라.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
7 3 6 1 7 2 3 1 4 4 5 1 2	2 1 0 0 0 0 0
15 6 2 3 1 6 10 8 7 13 12 6 5 11 14 6 3 7 9 6 15 6 8 3 4 6 13 6 11	3 2 2 1 1 1 1 0 0 0 0 0 0 0 0

이 페이지는 공백입니다

문제 K. 술 푸는 사람

시간 제한 1 초 메모리 제한 1024 MB



당신은 주점에서 일하게 되었는데 이 주점에서는 손님이 주문한 술을 조금 독특한 방식으로 준비한다. 점원은 **두 개의 국자**를 이용해 손님이 주문한 만큼의 술을 동이에 담아 손님에게 낸다. 두 국자는 각각 mL (밀리리터) 단위로 양의 정수인 용량을 가진다. 두 국자의 용량이 서로 같을 수도 있다.

동이에 술을 담는 작업을 시작할 때는 동이가 완전히 비어 있다. 술을 동이로 옮길 때 두 개의 국자 중 하나를 골라 다음 두 동작 중 하나를 할 수 있다.

- 술통에서 국자로 한 번 떠서 동이에 부어 넣는다. 이때 동이의 술의 양은 선택한 국자의 용량만큼 증가한다.
- 동이에서 국자로 한 번 떠서 밖으로 덜어 낸다. 이때 동이의 술의 양은 선택한 국자의 용량만큼 감소한다.

동이에 들어 있는 술의 양이 선택한 국자의 용량보다 적은 상태에서 동이에서 술을 뜨면 동이는 비게 되며, 동이의 술의 양은 음수가 될 수 없다. 또한 동이는 매우 커서 작업 도중 동이가 넘치는 일은 없다.

손님이 주문할 수 있는 술의 양은 1 mL부터 N mL까지의 모든 양의 정수 값이며 두 국자의 용량을 적절히 정하여 주문이 어떤 양이더라도 위에서 제시된 동작들을 통해 그 양을 정확히 맞출 수 있어야 한다.

영리한 당신은 최악의 경우에도 작업 횟수를 최대한 줄이기 위해 추가적으로 다음 조건을 만족하도록 만들고자 한다.

- 1 mL부터 N mL까지의 주문 중 술을 푸는 횟수가 **가장 많이 필요한 경우 그 가장 많이 푸는 횟수가 최소가 되도록** 두 국자의 용량을 정한다.

손님이 주문할 수 있는 최대 술의 양이 주어지면 술을 푸는 횟수가 가장 많이 필요할 때 몇 번 펴야 하는지, 그리고 두 국자의 용량은 얼마로 할지 답하라.

입력

손님이 주문할 수 있는 최대 술의 양을 mL 단위로 나타냈을 때의 값인 양의 정수 N 이 주어진다. ($1 \leq N \leq 2000000$)

출력

첫 번째 줄에 **최대 술 푸는 횟수**를 출력하여라.

두 번째 줄에 그때 **두 국자의 용량**을 mL 단위로 나타냈을 때의 값인 두 양의 정수를 공백으로 구분하여 출력하여라.

최대 술 푸는 횟수가 동일한 국자의 용량 쌍이 여러 가지 존재한다면 그중 아무거나 출력한다.

두 국자의 용량은 mL 단위로 각각 $2N$ 보다 작거나 같아야 한다. 이 조건을 만족시키며 답이 될 수 있는 국자 용량 쌍이 존재함이 보장된다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
5	2 1 4
10	3 2 5

두 번째 예제의 경우, 다음과 같이 10mL까지의 값을 3번 만에 모두 만들 수 있다.

$$1 = 5 - 2 - 2$$

$$2 = 2$$

$$3 = 5 - 2$$

$$4 = 2 + 2$$

$$5 = 5$$

$$6 = 2 + 2 + 2$$

$$7 = 2 + 5$$

$$8 = 5 + 5 - 2$$

$$9 = 5 + 2 + 2$$

$$10 = 5 + 5$$

문제 L. 정투상도

시간 제한 1 초 메모리 제한 1024 MB

당신에게 NMK 개의 정육면체 모양 나무 블록이 주어진다. NMK 는 '많다'의 은유가 아니라 정말로 NMK 개이다. 왜냐하면 블록을 놓을 수 있는 공간이 가로로 N 칸, 세로로 M 칸, 그리고 높이 K 칸이기 때문이다. 블록은 가로, 세로, 높이가 모두 1칸이며 각 격자 칸에 정확히 들어가도록 놓여야 한다. 이제 당신은 설계도에 맞춰 블록을 놓으려 한다.

당신은 입력으로 주어진 설계도를 그대로 출력할 생각에 잠시 신이 났지만 애석하게도 설계도가 제3각법으로 그려진 정면도, 평면도, 우측면도밖에 없다는 사실이 드러났다.

각 도면은 해당 방향에서 바라보았을 때 그 위치에 1개 이상의 블록이 있다면 **1**, 아무 블록도 없어 배경이 보이면 **0**으로 표시되어 있다. 같은 시선 방향에 여러 블록이 있어도 블록은 투명하지 않으므로 도면에는 똑같은 **1**로 표시된다.

보다 엄밀한 정의를 위해 공간의 각 칸을 3차원 직교좌표계를 이용해 (x, y, z) 로 나타내자. x 는 왼쪽에서 오른쪽 방향으로 커지며 $1 \leq x \leq N$ 이다. y 는 앞쪽에서 뒤쪽 방향으로 커지며 $1 \leq y \leq M$ 이다. z 는 아래쪽에서 위쪽 방향으로 커지며 $1 \leq z \leq K$ 이다.

원래의 블록 배치를 나타내는 배열 $B[x][y][z]$ 가 있을 때, 좌표 (x, y, z) 에 블록이 존재하면 $B[x][y][z] = 1$, 존재하지 않으면 $B[x][y][z] = 0$ 이라 하자.

정면도 D 의 경우, $D[z][x]$ 는 $B[x][y][z] = 1$ 인 y 가 존재하면 1, 아니면 0이다.

평면도 E 의 경우, $E[y][x]$ 는 $B[x][y][z] = 1$ 인 z 가 존재하면 1, 아니면 0이다.

우측면도 F 의 경우, $F[z][y]$ 는 $B[x][y][z] = 1$ 인 x 가 존재하면 1, 아니면 0이다.

가능한 블록 배치가 하나 이상 존재한다면 각 좌표 (x, y, z) 에 대해 배열 $A[x][y][z]$ 를 아래와 같이 정의한다.

- 모든 가능한 배치에서 블록이 존재하면 $A[x][y][z] = 2$ 이다.
- 모든 가능한 배치에서 블록이 존재하지 않으면 $A[x][y][z] = 0$ 이다.
- 블록이 존재하는 배치와 존재하지 않는 배치가 모두 가능하면 $A[x][y][z] = 1$ 이다.

입력

첫 번째 줄에 세 양의 정수 N, M, K 가 공백으로 구분되어 주어진다. ($1 \leq N, M, K \leq 100$)

그다음 K 개의 줄에는 정면도가 주어진다. $z = 1$ 인 줄부터 $z = K$ 인 줄까지 순서대로 주어지며, 각 줄은 길이 N 의 문자열이다. 정면도의 z 번째 줄, x 번째 문자는 $D[z][x]$ 를 나타낸다.

그다음 M 개의 줄에는 평면도가 주어진다. $y = 1$ 인 줄부터 $y = M$ 인 줄까지 순서대로 주어지며, 각 줄은 길이 N 의 문자열이다. 평면도의 y 번째 줄, x 번째 문자는 $E[y][x]$ 를 나타낸다.

그다음 K 개의 줄에는 우측면도가 주어진다. $z = 1$ 인 줄부터 $z = K$ 인 줄까지 순서대로 주어지며, 각 줄은 길이 M 의 문자열이다. 우측면도의 z 번째 줄, y 번째 문자는 $F[z][y]$ 를 나타낸다.

정면도, 평면도, 우측면도의 모든 문자는 **0** 또는 **1**이다.

출력

주어진 세 도면과 일치하는 블록 배치가 존재하지 않는다면 첫 번째 줄에 **Wrong design**을 출력하여야.

가능한 배치가 존재한다면 $M \times K$ 개의 줄에 걸쳐 배열 A 를 출력하여야. 각 줄은 **0, 1, 2**로만 이루어진 길이 N 의 문자열이어야 하며 $z = 1$ 에서 K 의 순서로 M 개의 줄을 순서대로 출력하여야. 즉 $A[x][y][z]$ 는 $M \times (z - 1) + y$ 번째 줄의 x 번째 문자이다.

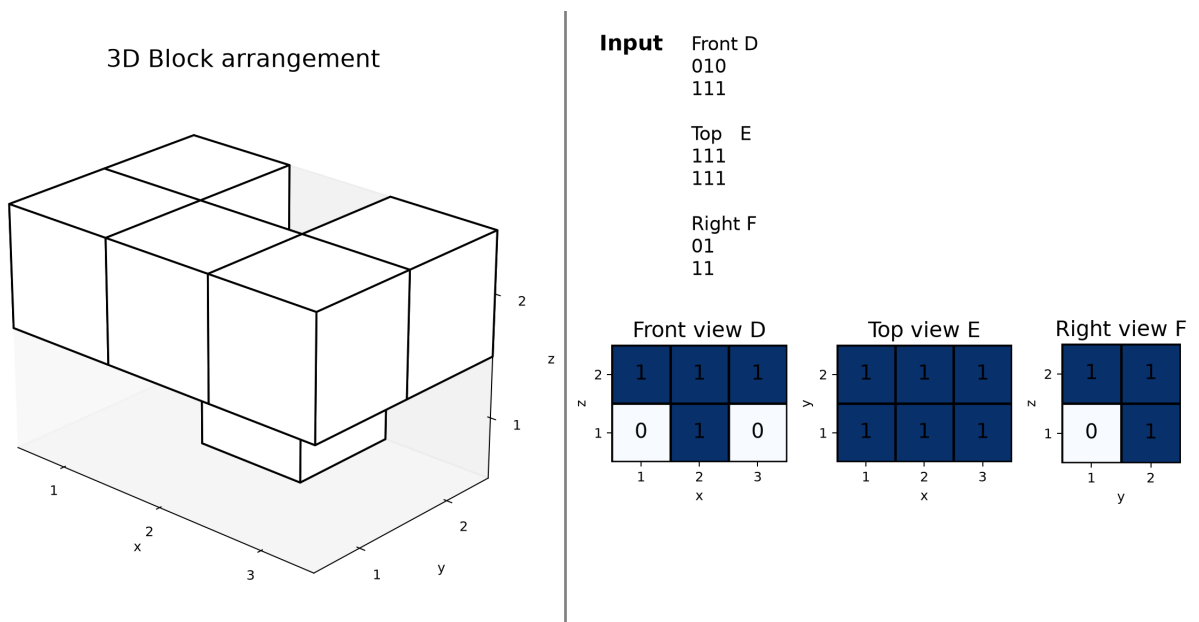
입출력 예시

표준 입력(stdin)	표준 출력(stdout)
3 2 2 010 111 111 111 01 11	000 020 222 212
2 2 2 10 00 01 00 00 01	Wrong design

노트

도면에 맞게 배치 가능하지만 하다면 모든 블록이 이웃하는 블록을 거쳐 연결되어야 할 필요는 없다. 마찬가지로 블록이 공중에 떠 있거나, 격자 공간에 여백이 있거나, 블록이 하나도 없는 경우도 가능함에 주의하라.

설계도와 일치하기만 한다면 NMK 개의 블록을 모두 사용할 필요는 없다.



첫 번째 예제에서 주어진 상황을 묘사한 그림이다.

문제 M. 쌀 재배하기와 퀴리

시간 제한 4 초 메모리 제한 1024 MB

엘리스는 유명한 농부다. 엘리스는 세로 N 칸, 가로 M 칸으로 되어 있는 논을 가지고 있으며 논에는 논배미들이 빈틈없이 배치되어 있다. 각 논배미는 세로 1칸 가로 1칸으로 되어 있는 작은 논이 상하좌우로 연결되어 있는 형태를 가지고 있다. 밥은 별로 유명하지 않은 농부다. 밥은 엘리스에게서 논의 일부를 빌려 쌀농사를 지으려고 한다. 이때 다음 조건들을 만족해야 한다.

- 밥이 빌릴 논은 직사각형 형태여야 하며, 엘리스가 가지고 있는 논 안에 속해야 한다.
- 밥은 위에서 i 번째, 왼쪽에서 j 번째 칸에 매직 쌀을 기르려고 한다. 그러므로 위에서 i 번째 왼쪽에서 j 번째 칸은 반드시 빌려야 한다.
- 논배미의 일부만을 빌릴 수 없다. 그러므로, 각 논배미마다 모든 칸이 밥이 빌리는 논에 속하거나, 모든 칸이 밥이 빌리는 논에 속하지 않아야 한다.
- 밥은 가난해서 최대한 작은 논을 빌리고 싶다. 위 조건들을 만족하는 논 중 가장 작은 크기의 논을 빌려야 한다. 만약 가장 작은 크기의 논이 여럿이라면 아무거나 빌려도 된다.

밥이 반드시 빌리고 싶은 칸 (i, j) 에 대한 퀴리가 Q 개 주어진다. 각 퀴리마다, 밥이 빌려야 하는 논을 구해주자!

입력

첫 번째 줄에 N, M 이 공백으로 구분되어 주어진다. ($1 \leq N, M \leq 400$)

두 번째 줄부터 N 개의 줄에 걸쳐 i 번째 줄에 $a_{i,1}, a_{i,2}, \dots, a_{i,M}$ 이 공백으로 구분되어 주어진다. ($1 \leq a_{i,j} \leq N \times M$)

$a_{i,j} = k$ 라는 것은 위에서 i 번째 왼쪽에서 j 번째 칸이 k 번 논배미가 차지하는 칸이라는 것을 의미하며, 같은 논배미에 속하는 칸은 상하좌우로 연결되어 있다.

그다음 줄에 Q 가 주어진다. ($1 \leq Q \leq N \times M$)

그다음 줄부터 Q 개의 줄에 걸쳐 i, j 가 공백으로 구분되어 주어진다. ($1 \leq i \leq N, 1 \leq j \leq M$)

출력

Q 개의 줄에 걸쳐, i 번째 줄에 U, L, D, R 을 출력하여라. 이는 i 번째 퀴리에서 밥이 위에서 U 번째 칸에서부터 D 번째 칸까지 왼쪽에서 L 번째 칸에서부터 R 번째 칸까지 빌려야 한다는 것을 의미한다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
6 7 1 1 1 1 1 1 1 1 2 2 2 5 5 1 1 2 3 3 5 5 1 1 2 4 4 4 5 1 1 6 6 6 6 5 1 1 1 1 1 1 1 1 6 2 2 2 5 3 3 4 4 5 3 1 1	2 2 5 6 2 2 5 6 3 3 3 4 4 3 4 5 5 2 5 5 1 1 6 7
5 5 1 1 1 1 1 2 2 2 2 1 2 1 1 2 1 2 1 2 2 1 2 1 1 1 1 2 3 3 4 3	1 1 5 5 1 1 5 5

문제 N. 매직 쌀 2

시간 제한 1 초 메모리 제한 1024 MB

여러분은 예비소집에 참가해 문제를 열심히 해결했는가? 아쉽게도, 예비소집에 출제된 매직 쌀 문제와 이 문제는 하나도 관련이 없다!

재호는 유명한 농부이다. 그는 유명한 농부인 앨리스와 밥으로부터 매직 쌀의 모종을 얻어왔다. 그도 벼농사를 열심히 하여 쌀을 많이 많이 팔고 엄청난 부자가 되고자 한다. 재호는 $N \times N$ 격자 모양의 논을 가지고 있으며, 각 칸에 벼를 하나씩 심을 수 있다. 위에서 x 번째, 왼쪽에서 y 번째 칸에 해당하는 논을 (x, y) 로 표현하자.

처음에 얻어온 모종벼는 $(1, 1)$ 에 심어져 있다. 벼를 더 심기 위해서는 다음을 반복해야 한다:

- 2×2 크기의 부분 격자를 이루는 네 칸의 논 중 한 칸에만 벼가 심어져 있는 부분격자를 선택한다.
- 나머지 세 칸의 논에 벼를 심는다.

재호가 엄청난 부자가 되기 위해서는 되도록 많은 칸에 벼를 심어야 한다. 하지만 위의 방법으로는 벼를 $3k+1$ 꼴의 개수 만큼만 심을 수 있으므로 논을 꽉 채우지 못할 수도 있다. 따라서, 재호는 빈칸이 **열 칸 이내**가 되도록 벼를 적당히 심고자 한다. 재호를 위해 벼를 적당히 심는 방법을 알려주자.

입력

입력은 하나 이상의 테스트 케이스로 이루어져 있다. 첫 번째 줄에 테스트 케이스의 개수 T 가 주어진다. ($1 \leq T \leq 150$)
두 번째 줄부터 T 개의 줄에 걸쳐, 그중 i 번째 줄에 각 테스트케이스에서 논을 가로세로 길이 N 이 주어진다. ($1 \leq N \leq 1000$)
모든 테스트케이스에서 N^2 의 합은 1000000을 넘지 않음이 보장된다.

출력

T 개의 테스트케이스에 대한 답을 줄바꿈으로 구분하여 출력하여야 한다. 아래는 각 테스트케이스에서 출력 양식이다.

첫 번째 줄에 부분격자를 선택할 횟수 M 을 출력하여야.

두 번째 줄부터 M 개의 줄에 걸쳐, 그중 i 번째 줄에 i 번째로 선택하는 2×2 부분격자 왼쪽 위 칸의 x 좌표와 y 좌표를 공백으로 구분하여 출력하여야.

가능한 답이 여러 가지 존재한다면 그중 아무거나 출력한다.

입출력 예시

표준 입력(stdin)	표준 출력(stdout)
4	0
1	1
2	1 1
3	2
3	1 1
	2 2
	0

첫 번째 테스트케이스에서, 존재하는 유일한 칸인 $(1, 1)$ 에 벼가 심어져 있으므로 빈칸은 없다.

세 번째 테스트케이스에서, 쌀을 두 번 심어서 총 7칸에 벼를 심었다. 빈칸은 두 칸이다. $N=3$ 인 경우 초기에 빈칸이 8개이므로, 네 번째 테스트케이스에서와 같이 벼를 심지 않아도 된다.

노트

$(1, 1)$ 에는 벼가 심어져 있으므로 빈칸으로 세지 않는다.

이 페이지는 공백입니다