APC

주최 아주대학교 AJOU UNIVERSITY | SW중심대학사업 National Center of Excellence in SW

주관 ANSi

후원 Samsung Software Membership SOLVED.AC STARTLINK

Ajou Univ. Programming Contest

2025 아주대학교 프로그래밍 경시대회



공식 문제지

Official Problemset

Division 2. Challenge Round

이 문제지는 **Division 2** 참가자를 위한 문제지입니다.

- 7문제가 정상적으로 인쇄되었는지 확인하십시오.
- 모든 문제의 메모리 제한은 1024MB로 동일합니다.
- 참가자는 대회 중 사전에 승인받은 하드카피를 참고할 수 있습니다.
- 본 대회의 문제는 출제자가 판단한 난이도 순으로 정렬되어 있습니다. 다만, 실제 체감 난이도는 참가자마다 다를 수 있으므로 모든 문제를 확인해 보시기를 권장합니다.
- 대회 중 문제에 관한 질문은 대회 페이지 내 질문 기능을 이용해주시기 바랍니다.
- 대회 중 퇴실은 불가합니다.

문항 목록 - Division 2.

문제		시간 제한
A	Acentuación del idioma español	1 초
B	클리크 조절	1 초
C	마지막 수강신청	1 초
D	카드 뭉치	1 초
E	Was It a Cat I Saw	1 초
F	가오리 그래프	1 초
G	양과 늑대	3 초

문제 A. Acentuación del idioma español

- 시간 제한 1초

문제

스페인어에는 단어의 발음을 명확히 하기 위해 강세가 사용된다. 강세는 단어마다 정확히 하나가 존재하며, 단어에 존재하는 모음(`a, e, i, o, u`)에만 붙을 수 있다.

만약 단어에 악센트 표시가 붙은 모음(`á, é, í, ó, ú`)이 존재한다면, 악센트 표시가 있는 모음에 강세를 준다. 이를 스페인어의 **불규칙적 강세**라고 한다. 단어에 악센트 표시가 붙은 모음이 단 하나도 존재하지 않는다면, 다음 **규칙적 강세** 규칙에 따라 강세의 위치가 정해진다:

- `n, s` 이외의 자음으로 끝나는 단어는 마지막 모음에 강세를 준다.
- 모음으로 끝나는 단어나 `n` 또는 `s`로 끝나는 단어는 뒤에서 두 번째 모음에 강세를 준다.

여기서 자음은 모음을 제외한 모든 알파벳을 의미한다.

영어 알파벳 소문자로만 이루어진 문자열 S가 주어진다. 문자열 S를 규칙적 강세 규칙에 따라 발음하고자 할 때, 어느 문자에 강세를 주어야 하는지 구하여라.

입력

첫 번째 줄에 영어 알파벳 소문자로만 이루어진 문자열 S가 주어진다. $(1 \leq |S| \leq 100)$

문자열 S의 문자에는 악센트 표시가 없음이 보장된다.

출력

강세가 붙는 문자가 왼쪽부터 몇 번째에 위치하는지 출력한다. 위치는 1부터 시작한다.

단, 강세의 위치를 정할 수 없는 경우는 `-1`을 출력한다.

예제

standard input	standard output
`casa`	2
`joven`	2
`cantar`	5
`can`	-1

1. 다음은 첫 번째 예제에 대한 설명이다:

   모음으로 끝나는 단어이기 때문에, 규칙적 강세의 2번 규칙을 적용하여 뒤에서 두 번째 모음인 2번째 문자에 강세가 붙는다.
2. 다음은 두 번째 예제에 대한 설명이다:

   `n`으로 끝나는 단어이기 때문에, 규칙적 강세의 2번 규칙을 적용받는다.
3. 다음은 세 번째 예제에 대한 설명이다:

   `n, s` 이외의 자음으로 끝나는 단어이기 때문에, 규칙적 강세의 1번 규칙을 적용받는다.
4. 다음은 네 번째 예제에 대한 설명이다:

   `n`으로 끝나는 단어이기 때문에 규칙적 강세의 2번 규칙을 적용받지만, 모음이 1개인 단어이기 때문에 위 강세 규칙에 따라 강세 위치를 정할 수 없다.

노트

- 실제 스페인어에서는 여러 개의 모음이 연속적으로 나오는 이중모음(diphthongs), 삼중모음(triphthongs)등의 경우 별도의 음절 분리 규칙을 적용받기 때문에 강세 위치가 달라질 수 있지만, 본 문제에서는 해당 규칙을 고려하지 않는다.
- 또한 실제 스페인어 알파벳은 영어의 26자에 더해 ñ 문자가 추가로 사용되지만, 본 문제에서는 영어 알파벳만을 고려한다.

문제 B. 클리크 조절

- 시간 제한 1초

문제

여기, 육군훈련소에 들어가게 된 훈련병 민우가 있다. 국방의 의무를 다하기 위해 입대한 민우는 입대하고 나서야 자신이 사격에 천재적인 재능이 있음을 발견하게 된다.

사격 훈련은 총 2회, 훈련마다 N발의 사격을 진행하게 된다. 각 사격 훈련이 끝난 뒤, 모든 훈련병은 발사된 총알들이 표적지에 꽂힌 위치를 확인한다. 표적지는 2차원 좌표계, 발사된 총알들이 꽂힌 위치는 2차원 좌표계의 점으로 생각할 수 있다.

두 번의 사격 훈련이 종료된 후, 민우는 훈련마다 자신이 쏜 총알들이 표적지에 꽂힌 위치가 첫 번째 사격과 비교해 정확히 x축 양의 방향으로 A만큼, y축 양의 방향으로 B만큼 평행이동한 것을 확인하였다. 민우를 도와 A와 B를 구해보자.

입력

첫 번째 줄에 각 사격 훈련마다 발사한 총알의 개수를 의미하는 양의 정수 N이 주어진다. $(1 \leq N \leq 200\,000)$

두 번째 줄부터 N줄에 걸쳐 첫 번째 사격 훈련에서 발사한 총알이 표적지에 꽂힌 위치의 좌표를 의미하는 정수 x와 y가 공백으로 구분되어 주어진다. $(-10^9 \leq x, y \leq 10^9)$

그다음 N줄에 걸쳐 두 번째 사격 훈련에서 발사한 총알이 표적지에 꽂힌 위치의 좌표를 의미하는 정수 x와 y가 공백으로 구분되어 주어진다. $(-10^9 \leq x, y \leq 10^9)$

출력

각 탄착군이 x축과 y축 양의 방향에 대하여 평행이동한 값을 의미하는 A와 B를 공백으로 구분하여 출력한다.

문제의 조건을 만족시키는 A와 B가 반드시 존재한다.

예제

standard input	standard output
3 0 0 1 0 1 1 3 2 3 3 2 2	2 2

문제 C. 마지막 수강신청

- 시간 제한 1초

문제

도현이는 어느덧 졸업을 앞두고 마지막 학기 수강신청을 하고 있다. 도현이는 졸업하기 전 학교에서 마지막으로 수강하고 싶은 후보 과목 N 개를 정리했다. 각 과목은 다음과 같은 정보를 가지고 있다:

- c_i: i 번째 과목을 수강하여 얻을 수 있는 학점을 나타낸다.
- s_i: 매주 i 번째 과목이 진행되는 강의 횟수를 나타낸다.
- $d_{i,j}$: 매주 i 번째 과목의 j 번째 강의가 열리는 요일을 나타낸다.
- $h_{i,j}$: 매주 i 번째 과목의 j 번째 강의가 열리는 시각을 나타낸다. 강의는 $h_{i,j}$ 시 정각에 시작해서 $h_{i,j}$ 시 59 분에 종료된다.

도현이는 이 강의들의 일부만으로 시간표를 구성했을 때, 졸업 요건인 M 학점을 수강할 수 있는지 알고 싶다. 수강하고자 하는 과목들 간 시간이 겹치지 않아야 하며, 이동시간 등은 고려하지 않는다.

입력

첫 번째 줄에 수강하고 싶은 후보 과목의 개수 N, 졸업을 위해 수강해야 하는 최소 학점 M 이 공백으로 구분되어 주어진다. $(1 \le N \le 10; 1 \le M \le 24)$

두 번째 줄부터 N 줄에 걸쳐 각각 후보 과목에 대한 정보가 주어진다. 각 줄에 c_i, s_i, $d_{i,1}, h_{i,1}, \ldots, d_{i,s_i}, h_{i,s_i}$ 가 공백으로 구분되어 주어진다. $(1 \le c_i \le 4; 1 \le s_i \le 3; 0 \le h_{i,j} \le 23;$ $d_{i,j}$ 는 `MON, TUE, WED, THU, FRI`중 하나의 값을 가진다.)

같은 과목의 강의 시간들은 오름차순으로 주어지며, 서로 다른 강의 시간이 겹치는 입력은 주어지지 않는다.

출력

후보 과목으로 시간표를 구성하여 M 학점 이상 수강할 수 있다면 `YES`, 아니라면 `NO`를 출력한다.

예제

standard input	standard output
3 6 3 2 MON 14 FRI 12 3 1 MON 14 2 2 TUE 16 FRI 12	NO
3 6 3 2 MON 14 FRI 12 2 1 MON 14 3 2 TUE 16 FRI 13	YES

노트

요일의 순서는 `MON` $<$ `TUE` $<$ `WED` $<$ `THU` $<$ `FRI` 로 정의되며, 각각 순서대로 월요일부터 금요일까지를 나타낸다.

문제 D. 카드 뭉치

- 시간 제한 1초

문제

앞면에 양의 정수가 적혀 있는 N 장의 카드 뭉치가 있다. 처음에 카드는 앞면이 위로 보이도록 일렬로 나열되어 있고, 왼쪽에서 i 번째 카드에 적혀있는 정수는 A_i 이다.

여러분은 아래의 규칙에 따라 카드 뭉치를 나눠 여러 개의 카드 뭉치로 만들어야 한다.

- 카드 뭉치를 나눌 때는 연속된 구간 단위로 분할하여 나눠야 한다.
- 카드 뭉치 내 카드의 순서는 바꿀 수 없다.
- 각 카드 뭉치에 속한 카드의 개수는 카드 뭉치의 가장 왼쪽에 있는 카드에 적힌 수보다 작거나 같아야 한다.
- 모든 카드는 정확히 하나의 카드 뭉치에 속해야 한다.

예를 들어 카드 뭉치에 4 장의 카드가 있고 왼쪽부터 카드에 $2, 3, 1, 1$ 이 적혀 있다고 하자. 위의 규칙에 따라 $[2, 3], [1], [1]$ 또는 $[2], [3, 1, 1]$ 로 카드 뭉치를 나눌 수 있지만 $[2, 1], [3, 1]$ 또는 $[2, 3, 1], [1]$ 로는 나눌 수 없다.

위의 규칙을 만족하면서 카드 뭉치의 개수가 최소가 되도록 나눴을 때, 그 개수를 구해보자.

입력

첫 번째 줄에 카드의 수 N 이 주어진다. $(1 \leq N \leq 3\,000)$

두 번째 줄에 각 카드에 적힌 양의 정수가 공백으로 구분되어 주어진다. i 번째 정수는 왼쪽에서 i 번째 카드에 적힌 정수를 의미한다. $(1 \leq A_i \leq N)$

출력

문제의 규칙을 만족하면서 카드 뭉치의 개수가 최소가 되도록 나눴을 때, 그 개수를 출력한다.

예제

standard input	standard output
4 2 3 1 1	2
5 1 2 3 4 5	3
6 6 5 4 3 2 1	1

문제 E. Was It a Cat I Saw

- 시간 제한 1초

문제



문제 제목의 문장을 거꾸로 읽어보자. 그렇다. 당신은 팰린드롬을 좋아한다.

팰린드롬(Palindrome)이란 앞으로 읽어도, 뒤로 읽어도 같은 문자열을 의미한다. 양의 정수 X가 주어질 때, 당신은 연산마다 아래의 연산 중 하나를 골라 시행할 수 있다.

- X에 1을 더한다.
- X에 1을 뺀다. 단, $X > 1$를 만족해야 한다.

X를 이진수로 표현한 문자열이 팰린드롬이 되도록 원하는 만큼 연산을 적용할 때, 필요한 연산의 최소 횟수를 구해보자. 이때 X를 이진수로 표현했을 때 앞쪽의 불필요한 0들(leading zero)은 무시한다. 예를 들어, $X = 9 = 1001_{(2)}$는 이진수로 표현했을 때 팰린드롬이지만, $X = 8 = 1000_{(2)}$는 이진수로 표현했을 때 팰린드롬이 아니다.

입력

첫 번째 줄에 테스트 케이스의 개수 T가 주어진다. $(1 \leq T \leq 30)$

두 번째 줄부터 T줄에 걸쳐 양의 정수 X가 주어진다. $(1 \leq X \leq 10^9)$

출력

각 테스트 케이스마다 X를 이진수로 표현한 문자열이 팰린드롬이 되도록 문제의 연산을 적용할 때, 필요한 연산의 최소 횟수를 출력한다.

예제

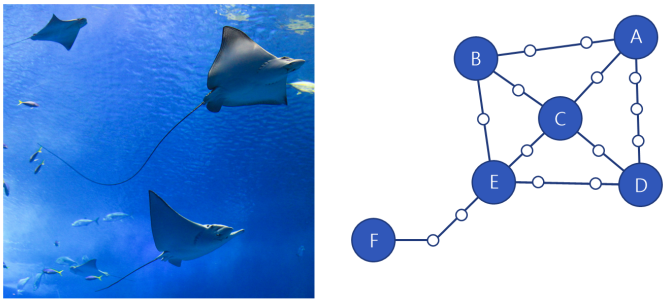
standard input	standard output
2 8 1	1 0

문제 F. 가오리 그래프

- 시간 제한 1초

문제

상혁이는 도서관에서 책을 빌려 보던 중, 책 한 장에 인상적인 가오리 그림이 그려진 페이지를 발견했다! 하지만 그림 위에 누군가 해 놓은 낙서를



왼쪽 : 가오리 그림, 오른쪽 : 가오리 그림을 따라서 그린 그래프

발견했고, 자세히 보니 단순한 낙서가 아니라 가오리 그림을 따라서 그린 그래프였다. 이 낙서에 감명을 받은 상혁이는 여기에 '가오리 그래프'라는 이름을 붙였다.

가오리 그래프는 N 개의 정점과 $N+3$ 개의 간선으로 이루어진 무방향 연결 그래프이며 아래와 같은 특징을 가지고 있다.

- 각 정점은 6 개의 핵심 정점과 $N-6$ 개의 일반 정점으로 구분된다. 핵심 정점은 머리(A), 왼쪽 날개(B), 중심(C), 오른쪽 날개(D), 아래쪽 날개(E), 꼬리(F) 이다.
- 핵심 정점 쌍 중 (A,B), (A,C), (A,D), (B,C), (B,E), (C,D), (C,E), (D,E), (E,F) 에 대해, 다른 핵심 정점을 거치지 않는 단순 경로가 유일하게 존재한다. 나머지 핵심 정점 쌍은 그러한 경로가 존재하지 않는다.
- 일반 정점은 핵심 정점 간 경로상에 위치한 정점으로, 서로 다른 2 개의 정점과 인접해 있다.

상혁이는 가오리 그래프의 각 정점에 1 번부터 N 번까지 번호를 매기고, 간선에 대한 정보를 메모했다.

시간이 흐르고, 상혁이는 우연히 자신이 작성했던 가오리 그래프에 대한 메모를 발견했다. 하지만 상혁이는 책을 반납한 뒤여서 가오리 그래프의 원래 모습을 떠올릴 수 없었다. 하지만 오른쪽 날개(D)의 번호는 왼쪽 날개(B)의 번호보다 크다는 점은 알고 있다. 상혁이를 위해 가오리 그래프의 핵심 정점이 무엇인지 알려주자.

입력

첫 번째 줄에는 정점의 수 N 이 주어진다. $(6 \le N \le 50)$

다음 줄부터 $N+3$ 개의 줄에 걸쳐 간선에 대한 정보로, 간선이 연결하는 두 정점 U, V 가 공백으로 구분되어 주어진다. $(1 \le U, V \le N; U \neq V)$

입력으로 주어지는 그래프는 가오리 그래프임이 보장된다.

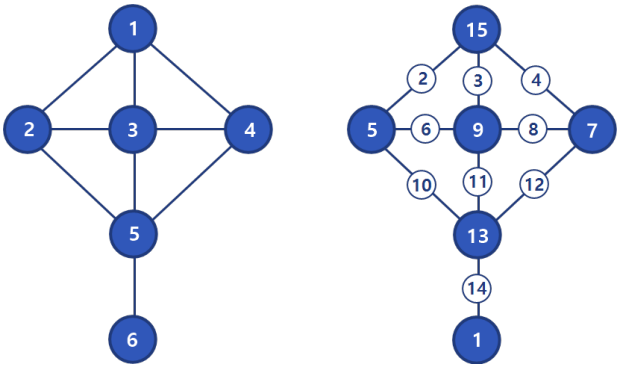
출력

첫 번째 줄에 머리, 왼쪽 날개, 중심, 오른쪽 날개, 아래쪽 날개, 꼬리를 의미하는 핵심 정점 A, B, C, D, E, F 의 번호를 공백으로 구분하여 출력한다. 입력으로 주어지는 조건에 해당하는 가오리 그래프가 유일하게 존재함을 증명할 수 있다.

예제

standard input	standard output
6 1 2 1 3 1 4 2 3 2 5 3 4 3 5 4 5 5 6	1 2 3 4 5 6
15 14 13 15 2 2 5 3 15 1 14 3 9 7 4 5 6 9 6 13 11 9 8 4 15 8 7 5 10 10 13 9 11 7 12 12 13	15 5 9 7 13 1

힌트



왼쪽 : 예제 1의 가오리 그래프, 오른쪽 : 예제 2의 가오리 그래프

문제 G. 양과 늑대

- 시간 제한 3초

문제

양치기 소년은 외딴 목장에서 N 마리의 양과 M 마리의 늑대를 기르고 있다.

각 양은 1 번부터 N 번까지 번호가 붙어 있으며 i 번 양의 크기는 A_i 이다. 각 늑대 또한 1 번부터 M 번까지 번호가 붙어 있으며 j 번 늑대의 크기는 B_j 이다.

양치기 소년은 자신이 기르고 있는 동물들을 관리하기 위해 우리를 만들고자 한다. 양치기 소년이 만든 우리는 최대 2 마리의 동물이 들어갈 수 있으며, 우리에 들어간 동물의 크기의 합은 K 를 초과할 수 없다. 각 우리에는 같은 종류의 동물만 들어갈 수 있지만, 예외적으로 양치기 소년이 감시할 수 있는 우리 하나에는 양과 늑대가 같이 들어갈 수 있다. 또한 모든 동물의 크기는 K 를 넘지 않는다.

모든 동물을 우리에 넣어야 할 때, 양치기 소년이 만들어야 하는 우리는 최소 몇 개인지 구해보자.

입력

첫 번째 줄에 N, M, K 가 공백으로 구분되어 주어진다. $(1 \le N, M \le 500\,000; 1 \le K \le 10^9)$

두 번째 줄에 양의 크기를 의미하는 $A_1, A_2, \cdots, A_N$ 가 공백으로 구분되어 주어진다. $(1 \le A_i \le K)$

세 번째 줄에 늑대의 크기를 의미하는 $B_1, B_2, \cdots, B_M$ 가 공백으로 구분되어 주어진다. $(1 \le B_j \le K)$

입력으로 주어지는 모든 값은 정수이다.

출력

첫 번째 줄에 양치기 소년이 만들어야 하는 우리의 최소 개수를 출력한다.

예제

standard input	standard output
4 4 8 3 7 4 2 7 4 7 8	6
5 4 2 1 1 1 1 1 1 1 1 1	5
1 1 5 2 3	1

마지막 페이지입니다.