

## 딱지놀이

두 어린이 A, B가 딱지놀이를 한다. 딱지놀이 규칙은 다음과 같다. 두 어린이는 처음에 여러 장의 딱지를 가지고 있고, 매 라운드마다 각자 자신이 가진 딱지 중 하나를 낸다. 딱지에는 별(★), 동그라미(●), 네모(■), 세모(▲), 네 가지 모양 중 하나 이상의 모양이 표시되어 있다. 두 어린이가 낸 딱지 중 어느 쪽이 더 강력한 것인지는 다음 규칙을 따른다.

- 만약 두 딱지의 별의 개수가 다르다면, 별이 많은 쪽의 딱지가 이긴다.
- 별의 개수가 같고 동그라미의 개수가 다르다면, 동그라미가 많은 쪽의 딱지가 이긴다.
- 별, 동그라미의 개수가 각각 같고 네모의 개수가 다르다면, 네모가 많은 쪽의 딱지가 이긴다.
- 별, 동그라미, 네모의 개수가 각각 같고 세모의 개수가 다르다면, 세모가 많은 쪽의 딱지가 이긴다.
- 별, 동그라미, 네모, 세모의 개수가 각각 모두 같다면 무승부이다.

예를 들어, 두 어린이 A, B가 낸 딱지가 다음 그림과 같다고 하자.



위 규칙을 따르면 A의 딱지는 별 하나

를 가지고 있고 B의 딱지는 별이 없으므로 승자는 A이다. 위의 그림이 라운드 1의 상황이었고, 라운드 2, 3, 4, 5의 상황이 아래 표와 같을 때, 라운드 2, 3, 4의 승자는 각각 B, B, A이며, 라운드 5에서는 무승부가 되어 이를 D로 표현하였다.

라운드	A의 딱지	B의 딱지	결과
1	★	●●■▲	A
2	■★●■▲	★●●▲	B
3	●■▲▲	■●■▲	B
4	★●■▲	★●■	A
5	★★■●▲	★■★▲●	D

별, 동그라미, 네모, 세모를 각각 숫자 4, 3, 2, 1로 표현한다. 예를 들어, 라운드 1의 경우 어린이 A가 낸 딱지의 그림 ★는 4로 표현할 수 있고, 어린이 B가 낸 딱지의 그림 ●●■▲는 3 3 2 1로 표현할 수 있다.

라운드의 수  $N$ 과 두 어린이가 순서대로 내는 딱지의 정보가 주어졌을 때, 각 라운드별로 딱지놀이의 결과를 구하는 프로그램을 작성하시오.

소스파일의 이름은 card.c 또는 card.cpp를 권장하지만, 서버에 제출하는 데는 다른 이름도 상관없다.

### 입력 형식

표준 입력으로 다음 정보가 주어진다. 첫 번째 줄에는 딱지놀이의 총 라운드 수를 나타내는 자연수  $N$ 이 주어진다.  $N$ 은 1 이상 1,000 이하이다. 다음 줄에는 라운드 1에서 어린이 A가 내는

딱지에 나온 그림의 총 개수  $a$ 가 주어진다.  $a$ 는 1 이상 100 이하이다. 뒤따라 나오는  $a$ 개의 정수는 어린이 A가 낸 딱지의 그림을 나타내는데, 각각 4, 3, 2, 1 중 하나의 값이다. 4, 3, 2, 1의 순서대로 주어지지 않을 수 있음에 주의하라. 다음 줄에는 라운드 1에서 어린이 B가 내는 딱지에 나온 그림의 총 개수  $b$ 가 주어진다.  $b$ 도 1 이상 100 이하이다. 뒤따라 나오는  $b$ 개의 정수는 어린이 B가 낸 딱지의 그림을 나타내는데, 역시 4, 3, 2, 1 중 하나의 값이다. 역시 4, 3, 2, 1의 순서대로 주어지지 않을 수 있음에 주의하라. 다음 두 줄에는 라운드 2에서 어린이 A, B가 낸 딱지의 그림들을 같은 식으로 표현한다. 위와 같은 식으로 매 라운드마다 두 어린이가 낸 딱지의 정보는 두 줄에 표현되며,  $N$  라운드의 딱지 정보는 차례대로 총  $2N$ 개의 줄에 주어진다.

### 출력 형식

표준 출력으로 총  $N$ 줄을 출력한다. 출력의  $i$ 번째 ( $1 \leq i \leq N$ ) 줄에 정확히 한 글자를 출력하는데, 출력하는 글자는 A, B, D 중 하나로 라운드  $i$ 의 결과를 나타낸다. 각 라운드의 결과는 A가 승자라면 A, B가 승자라면 B, 무승부라면 D이다.

### 부분문제의 제약 조건

- **부분문제 1:** 전체 점수 100점 중 2점에 해당하며 입력 예시로 주어진 입력만 존재한다.
- **부분문제 2:** 전체 점수 100점 중 13점에 해당하며  $a=1$ 이고  $b=1$ 로 가정한다.
- **부분문제 3:** 전체 점수 100점 중 17점에 해당하며 모든 딱지에는 별모양 그림만 나온다.
- **부분문제 4:** 전체 점수 100점 중 68점에 해당하며 원래의 제약조건 이외에 아무 제약조건이 없다.

### 입력과 출력의 예

#### 입력(1)

```
5
1 4
4 3 3 2 1
5 2 4 3 2 1
4 4 3 3 1
4 3 2 1 1
4 2 3 2 1
4 4 3 2 1
3 4 3 2
5 4 4 2 3 1
5 4 2 4 1 3
```

#### 출력(1)

```
A
B
B
A
D
```

#### 입력(2)

```
4
4 4 3 2 1
4 1 4 3 2
4 3 3 2 1
4 4 3 3 3
4 4 3 3 3
4 3 4 3 2
4 3 2 1 1
3 3 2 1
```

#### 출력(2)

```
D
B
A
A
```

## 방 배정하기

정보 초등학교 6학년 여학생들은 단체로 2박 3일 수학여행을 가기로 했다. 학생들이 묵을 숙소에는 방의 정원(방 안에 있는 침대 수)을 기준으로 세 종류의 방이 있으며, 같은 종류의 방들이 여러 개 있다. 정보 초등학교에서는 학생들에게 이 방들을 배정하되, 배정된 모든 방에 빈 침대가 없도록 하고자 한다.

예를 들어, 방의 종류가 5인실, 9인실, 12인실이고 6학년 여학생 전체가 113명이라면, 5인실 4개, 9인실 5개, 12인실 4개를 예약하면 각 방에 남은 침대 없이 배정이 가능하다. 또한 12인실은 사용하지 않고 5인실 10개와 9인실 7개만 사용하는 것도 가능하다. 그러나 방의 종류가 3인실, 6인실, 9인실이고 6학년 여학생 전체가 112명이라면 빈 침대 없이 방을 배정하는 것은 불가능하다.

방의 정원을 나타내는 서로 다른 세 자연수와 전체 학생 수를 나타내는 자연수 하나가 주어졌을 때, 배정된 모든 방에 빈 침대가 없도록 방 배정이 가능한지를 결정하는 프로그램을 작성하시오. 단, 세 종류의 방은 모두 충분한 개수가 있다고 가정하며, 위의 예에서와 같이 세 종류의 방을 모두 활용하지 않고 한 종류 또는 두 종류의 방만 이용하여 배정하는 것도 허용한다.

소스파일의 이름은 `room.c` 또는 `room.cpp`를 권장하지만, 서버에 제출하는 데는 다른 이름도 상관없다.

## 입력 형식

표준 입력으로 방의 정원을 나타내는 서로 다른 세 자연수  $A, B, C$  ( $1 \leq A < B < C \leq 50$ )와 전체 학생 수를 나타내는 자연수  $N$  ( $1 \leq N \leq 300$ )이 공백으로 분리되어 한 줄에 주어진다.

## 출력 형식

빈 침대 없이 배정이 가능할 경우 표준 출력으로 1을, 불가능할 경우 0을 출력한다.

## 부분문제의 제약 조건

- **부분문제 1:** 전체 점수 100점 중 3점에 해당하며 입력 예시로 주어진 입력만 존재한다.
- **부분문제 2:** 전체 점수 100점 중 5점에 해당하며  $A = 1$ 이다.
- **부분문제 3:** 전체 점수 100점 중 14점에 해당하며  $B, C$ 는  $A$ 의 배수이다.
- **부분문제 4:** 전체 점수 100점 중 78점에 해당하며 원래의 제약조건 이외에 아무 제약조건이 없다.

## 입력과 출력의 예

입력(1)

5 9 12 113

출력(1)

1

입력(2)

3 6 9 112

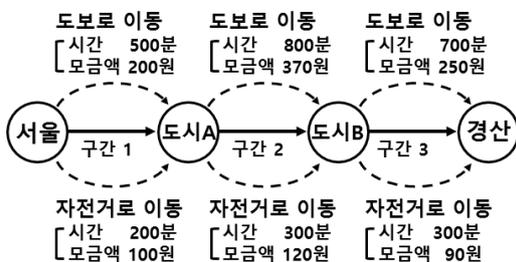
출력(2)

0

## 서울에서 경산까지

배우 한정을 씨는 이번 여름에 서울에서 경산까지 자선 여행을 하면서 모금 활동을 진행할 계획이다. 자선 여행에서 거쳐 가게 될 도시의 개수와 순서는 미리 정해져 있으며, 자선 여행은 서울에서 시작하여 각 도시를 정해진 순서대로 단 한 번씩 방문한 후 경산에서 끝난다. 서울을 제외한 도시의 개수를  $N$ 이라 하자. 이 때 서울에서 두 번째 도시까지 가는 구간을 구간 1, 두 번째 도시부터 세 번째 도시까지 가는 구간을 구간 2와 같이 부르기로 하며, 마지막 목적지인 경산에 도착하는 구간을 구간  $N$ 이라 하자. 즉, 구간의 전체 개수는  $N$ 이다. 구간 사이의 이동은 도보 혹은 자전거 어느 한 쪽을 이용하게 되는데, 각 구간에는 도보로 이동할 때 걸리는 시간(분), 이 때 얻게 되는 모금액(원), 자전거로 이동할 때 걸리는 시간(분), 이 때 얻게 되는 모금액(원)이 정해져 있다.

예를 들어, 서울과 경산 사이에 2개의 도시가 있는 다음과 같은 경우( $N=3$ )를 생각해 보자.



인접한 도시 사이를 도보로 이동하는지 자전거로 이동하는지에 따라 전체 모금액이나 걸리는 시간에 차이가 생기게 된다. 한정을 씨는 전체 모금액을 가능

한 많이 얻는 방법을 찾고 싶어 한다. 위의 예에서는 시간이 충분하다면 모든 구간을 도보로 이동하는 것이 모금액을 최대로 하는 방법이며, 모금액은  $200+370+250 = 820$ 원, 여행에 걸리는 시간은  $500+800+700 = 2,000$ 분이다.

그러나 한정을 씨는 바쁜 스케줄로 인해 자선 여행을 위해 보낼 수 있는 시간이  $K$ 분( $K$ 는 자연수)으로 한정되어 있다. 위의 예에서 만약  $K=1,650$ 이라면, 1, 2번 구간은 도보로 이동하고 3번 구간은 자전거로 이동하여 모금액을 660원으로 하는 것이 가장 좋은 방법이며, 이 때 걸리는 시간은 1,600분이다.

위와 같이 각 구간별로 도보 및 자전거로 이동하는 경우 걸리는 시간과 모금액이 주어질 때, 제한시간 이내로 서울에서 경산까지 여행하면서 모금할 수 있는 최대 금액을 찾는 프로그램을 작성하시오. (제한시간 이내에 여행하는 방법은 항상 존재한다.)

소스파일의 이름은 `travel.c` 또는 `travel.cpp`를 권장하지만, 서버에 제출하는 데는 다른 이름도 상관없다.

## 입력 형식

표준 입력으로 다음 정보가 주어진다. 첫 번째 줄에는 두 자연수  $N$ 과  $K$ 가 공백으로 분리되어 주어진다( $3 \leq N \leq 100$ ,  $0 < K \leq 100,000$ ). 두 번째 줄에는 구간 1을 도보로 이동할 때 걸리는 시간(분), 이 때 얻게 되는 모금액(원), 자전거로 이동할 때 걸리는 시간(분), 이 때 얻게 되는 모금액(원)을 나타내는 네 개의 자연수가 차례로 공백으로 분리되어 주어진다. 세 번째 줄부터

$N+1$ 번 째 줄도 마찬가지로 형식으로 각 줄마다 네 개의 자연수가 주어지며, 입력은 총  $N+1$ 줄로 구성된다. 두 번째 줄부터  $N+1$ 번 째 줄에 주어지는 숫자들 중 시간을 나타내는 숫자(각 줄의 첫 번째, 세 번째 숫자)는 10,000 이하의 자연수, 모금액을 나타내는 숫자(각 줄의 두 번째, 네 번째 숫자)는 1,000,000 이하의 자연수들이다.

### 출력 형식

표준 출력으로  $K$ 분 이내로 여행하면서 모금할 수 있는 최대 금액을 출력한다. ( $K$ 분 이내에 여행하는 방법은 항상 존재한다.)

### 부분문제의 제약 조건

- **부분문제 1:** 전체 점수 100점 중 29점에 해당하며  $N$ 의 범위를  $N \leq 20$ 으로 제한한다.
- **부분문제 2:** 전체 점수 100점 중 71점에 해당하며 원래의 제약조건 이외에 아무 제약조건이 없다.

### 입력과 출력의 예

입력(1)

```
3 1650
500 200 200 100
800 370 300 120
700 250 300 90
```

출력(1)

```
660
```

입력(2)

```
4 3000
1000 2000 300 700
1100 1900 400 900
900 1800 400 700
1200 2300 500 1200
```

출력(2)

```
5900
```

입력(3)

```
3 600
500 150 200 1000
100 835 200 324
200 125 300 900
```

출력(3)

```
2735
```

## 줄서기

$N$ 명의 학생들이 앞뒤로 일렬로 서 있다. 각 학생은 1부터  $N$ 까지 서로 다른 번호가 적힌 카드들 중 하나를 가지고 있다. 학생들에게서 자신보다 뒤에 서 있으면서 더 작은 번호의 카드를 가진 학생들의 명단을 **하나도 빠짐없이 모두** 받았다. 이 명단을 통해 학생들이 가지고 있는 카드의 번호를 알아내려고 한다.

예를 들어, 일렬로 서 있는 5명의 학생들을 앞에서부터 순서대로 “학생1, 학생2, 학생3, 학생4, 학생5”라고 하고, 학생들에게 받은 명단을 통해 다음과 같이 5개의 순서쌍이 만들어졌다고 하자. 순서쌍  $(X, Y)$ 는 학생  $Y$ 가 학생  $X$ 보다 뒤에 있으면서 더 작은 번호를 가지고 있다는 것을 의미한다.

(1,2), (1,5), (3,4), (3,5), (4,5)

이 자료를 분석하면 학생1, 학생2, 학생3, 학생4, 학생5는 각각 3, 1, 5, 4, 2가 적힌 카드를 가지고 있음을 알 수 있다.

다른 예로 5명 학생들에게 받은 명단으로 다음과 같은 6개의 순서쌍이 만들어졌다고 하자.

(1,2), (1,3), (1,5), (2,5), (3,4), (3,5)

이 경우, 학생들이 잘못된 명단을 제시한 것이다. 순서쌍 (2,5)에 의하면 학생2는 학생5보다 큰 번호의 카드를 가지고 있다. 그런데 만일 학생4의 카드가 학생

5의 카드보다 작은 번호라면 순서쌍 (2,4)가 존재해야 하고, 반대로 학생4의 카드가 학생5의 카드보다 큰 번호라면 순서쌍 (4,5)가 존재해야 한다. 그런데 둘 다 존재하지 않기 때문에 학생들이 잘못된 명단을 제시한 것이다.

학생들로부터 받은 명단으로 만들어진 순서쌍을 입력으로 받아, 학생들이 가지고 있는 카드 번호를 알아내는 프로그램을 작성하라.

소스파일의 이름은 line.c 또는 line.cpp를 권장하지만, 서버에 제출하는 데는 다른 이름도 상관없다.

## 입력 형식

표준 입력으로 다음 정보가 주어진다. 첫 번째 줄에는 학생 수  $N(1 \leq N \leq 100,000)$ 과 순서쌍의 수  $M(0 \leq M \leq 1,000,000)$ 이 공백으로 분리되어 주어진다. 일렬로 서 있는 학생들을 순서대로 학생1, 학생2, ..., 학생 $N$ 이라고 하자. 다음  $M$ 개의 각 줄에는 두 개의 자연수  $X$ 와  $Y$ 가 공백으로 분리되어 주어진다. 이것은 학생  $Y$ 가 학생  $X$ 보다 더 작은 번호가 적힌 카드를 가지고 있다는 것을 의미하는 순서쌍이다 ( $1 \leq X < Y \leq N$ ). 입력에 중복된 순서쌍은 없다.

## 출력 형식

표준 출력으로, 주어진 순서쌍을 통해 학생들이 가지고 있는 카드 번호를 알 수 있으면 학생들이 서 있는 순서대로 카드번호를 공백으로 분리하여 출력한다. 그렇지 않으면 -1을 출력한다.

### 부분문제의 제약 조건

- **부분문제 1:** 전체 점수 100점 중 7점에 해당하며  $N \leq 9$  이다.
- **부분문제 2:** 전체 점수 100점 중 21점에 해당하며  $N \leq 100$ 이다.
- **부분문제 3:** 전체 점수 100점 중 30점에 해당하며  $N \leq 5,000$ 이다.
- **부분문제 4:** 전체 점수 100점 중 42점에 해당하며 원래의 제약조건 이외에 아무 제약조건이 없다.

### 입력과 출력의 예

입력(1)

```
5 5
1 2
1 5
3 4
3 5
4 5
```

출력(1)

```
3 1 5 4 2
```

입력(2)

```
5 6
1 2
1 3
1 5
2 5
3 4
3 5
```

출력(2)

```
-1
```