

문제 1. 수도

입력 파일: standard input
출력 파일: standard output
시간 제한: 5 seconds
메모리 제한: 512 megabytes

JOI 왕국에는 1번부터 N 번까지 번호가 붙어 있는 N 개의 마을이 있다. 이 마을 사이를 연결하는 $N - 1$ 개의 도로가 있다. i 번째 ($1 \leq i \leq N - 1$) 도로는 A_i 번 마을과 B_i 번 마을 사이를 연결한다. 모든 도로는 양방향으로 지날 수 있다. 어떤 마을에서도 다른 모든 마을 까지 몇 개의 도로를 거치면 갈 수 있다.

현재, JOI 왕국은 1번부터 K 번까지 번호가 붙어있는 K 개의 도시로 나뉘어 있다. j 번 마을은 C_j 번 도시에 속한다. 모든 도시에는 적어도 하나의 마을이 속해 있다.

K 이사장은 JOI 왕국의 왕이다. 그는 정확히 하나의 도시를 수도로 고를 것이다. 안보 문제로, 수도는 다음과 같은 조건을 만족해야 한다.

해당 도시에 있는 어떤 두 마을 사이에도, 그 도시에 포함된 마을만 거쳐서 오가는 것이 가능하다.

하지만, K 이사장은 현재 도시가 수도를 고를 수 없는 구조 일 수 있다는 생각을 했다.

이 문제를 해결하기 위해서 K 이사장은 도시를 합칠 것이다. 정확히는 다음 조작을 할 수 있다.

$1 \leq x \leq K, 1 \leq y \leq K$ 와 $x \neq y$ 를 만족하는 x, y 를 골라서 y 번 도시에 속하는 모든 마을을 x 번 도시에 속하도록 변경한다.

도시를 합치는 데에는 매우 큰 비용이 들기 때문에, K 이사장은 도시를 합치는 횟수를 최소화 하여 수도를 고르고 싶다.

JOI왕국의 마을과 도로의 구조, 도시와 어떤 마을이 어떤 도시에 속해 있는지가 주어질 때, 도시를 합치는 횟수의 최솟값을 구하여라.

입력 형식

표준 입력에서 다음과 같은 형식으로 주어진다. 모든 값은 정수이다.

N K

A_1 B_1

\vdots

A_{N-1} B_{N-1}

C_1

\vdots

C_N

출력 형식

표준 출력에 하나의 줄을 출력하여라. 수도를 고르기 위한 도시를 합치는 횟수의 최솟값을 출력하여라.

제한

- $1 \leq N \leq 200\,000$.
- $1 \leq K \leq N$.
- $1 \leq A_i \leq N$ ($1 \leq i \leq N - 1$).

- $1 \leq B_i \leq N$ ($1 \leq i \leq N - 1$).
- 어떤 마을에서도 다른 모든 마을 까지 몇 개의 도로를 거치면 갈 수 있다.
- $1 \leq C_j \leq K$ ($1 \leq j \leq N$).
- 모든 k 에 ($1 \leq k \leq K$) 대해, $C_j = k$ 를 만족하는 정수 j 가 ($1 \leq j \leq N$) 존재한다.

서브태스크 1 (1 점)

- $N \leq 20$

서브태스크 2 (10 점)

- $N \leq 2\,000$

서브태스크 3 (30 점)

- 모든 마을은 도로로 직접 연결된 마을이 최대 2개이다.

서브태스크 4 (59 점)

추가 제한조건이 없다.

예제

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

이 예제에서, $(x, y) = (1, 3)$ 을 골라 3번 도시를 1번 도시에 합친다. 이 때, 1번 도시를 수도로 고를 수 있다. 처음에 어떤 도시도 수도로 고를 수 없다. 그렇기 때문에 도시를 합치는 횟수의 최솟값은 1이다.

이 예제는 서브태스크 1, 2와 4의 조건을 만족한다.

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

이 예제는 서브태스크 1, 2, 3과 4의 조건을 만족한다.

문제 2. 전설의 경단 요리사

출력 파일: Output Only

당신은 경단이라고 부르는 떡을 만드는 명인이다. 이제 당신은 경단을 꼬치에 꽂으려고 하고 있다.

경단은 가로 R 행, 세로 C 열의 격자 모양으로 배치되어 있다. 각 격자 칸에는 하나의 경단이 놓여 있다. 각 경단은 분홍색 (P), 하얀색 (W) 혹은 초록색 (G) 이다. 당신은 수직 (위에서 아래), 수평 (왼쪽에서 오른쪽), 혹은 사선 (왼쪽 위에서 오른쪽 아래 혹은 오른쪽 위에서 왼쪽 아래) 방향으로 연속된 세 개의 연속된 격자간에 있는 경단을 고를 것이다. 그리고 세 경단을 꼬치에 꽂는 것으로 **경단꼬치**를 만들 수 있다. 예를 들어, 세 경단을 수직방향으로 고르면 위, 가운데, 아래 순서 혹은 아래, 가운데, 위 순서로 꼬치를 꽂아 만들 수 있다. 중앙, 아래, 위 혹은 아래, 위, 중앙 같은 순서로 꼬치를 꽂을 수는 없다. 또한 같은 경단을 두 개 이상의 서로다른 꼬치에 꽂을 수도 없다.

경단꼬치 중에서 경단의 색이 분홍색, 하얀색, 초록색 혹은 초록색, 하얀색, 분홍색 순서인 경단을 **아름다운 경단꼬치**라고 부른다. 당신은 아름다운 경단꼬치를 최대한 많이 만들고 싶다.

몇개의 아름다운 경단꼬치를 만드는 것이 가능한가.

입력 형식

표준 입력에서 다음과 같은 형식으로 주어진다. 모든 값은 정수이다.

R C

D_1

\vdots

D_R

D_i 는 ($1 \leq i \leq R$) P, W, G로 이루어진 길이 C 의 문자열이다. D_i 의 j 번째 ($1 \leq j \leq C$) 문자는 위에서 i 번째 행, 왼쪽에서 j 번째 열에 들어 있는 경단의 색을 의미한다.

출력 형식

다음과 같은 형식으로 출력하여야.

S_1

\vdots

S_R

S_i 는 ($1 \leq i \leq R$) P, W, G, |, -, \, /로 이루어진 길이 C 의 문자열이다. S_i 의 j 번째 ($1 \leq j \leq C$) 문자는 위에서 i 번째 행, 왼쪽에서 j 번째 열에 들어있는 경단이 어떤 방식으로 꼬치에 꽂혀 있는가를 의미한다.

S_i 의 j 번째 문자가

- 해당하는 경단과 그 위와 아래의 경단을 하나의 꼬치로 만들 경우에는, 문자 |,
- 해당하는 경단과 그 왼쪽과 오른쪽의 경단을 하나의 꼬치로 만들 경우에는, 문자 -,
- 해당하는 경단과 그 왼쪽 위와 오른쪽 아래의 경단을 하나의 꼬치로 만들 경우에는, 문자 \,
- 해당하는 경단과 그 오른쪽 위와 왼쪽 아래의 경단을 하나의 꼬치로 만들 경우에는, 문자 /,
- 그 이외의 경우에는, 해당하는 격자에 있는 경단의 색을 의미하는 P, W, G (즉, D_i 의 j 번째 문자)

이다.

주어진 각 입력 데이터에 대응하는 출력 데이터만 제출하여야.

제한

- $3 \leq R \leq 500$.
- $3 \leq C \leq 500$.
- D_i 는 ($1 \leq i \leq R$) P, W, G로 이루어진 길이 C 의 문자열이다.

배점

득점은 다음과 같이 계산된다.

각 테스트 케이스에 대해 배점, 기본 점수, 중간 점수, 만점 점수가 다음과 같이 정해져 있다.

테스트 케이스	배점	기본 점수	중간 점수	만점 점수
01	15	44000	47000	47220
02	15	39000	41700	41980
03	15	45000	51000	51390
04	15	18000	19000	19120
05	20	43000	48200	48620
06	20	44000	46000	46500

제출한 출력에서 만들수 있는 아름다운 경단꼬치의 수를 N , 기본 점수를 X , 중간 점수를 Y , 만점 점수를 S , 배점을 S 라고 할 때, 각 테스트 케이스의 점수는 다음과 같이 계산된다.

- $N < X$ 일 때, 0점.
- $X \leq N < Y$ 일 때, $\frac{N - X}{2(Y - X)} \times S$ 점.
- $Y \leq N < Z$ 일 때, $\left(\frac{1}{2} + \frac{N - Y}{2(Z - Y)}\right) \times S$ 점.
- $Z \leq N$ 일 때, S 점.

각 테스트 케이스의 점수 합의 소숫점 아래를 반올림한 값이 이 문제에 대한 전체 점수이다.

단, 출력이 틀린 경우, 즉 출력의 l, -, \, / 문자에 대응되는 아름다운 경단꼬치를 만들 수 없는 경우, 출력에 포함된 P, W, G가 입력과 모순되는 경우, 혹은 출력이 지정된 형식과 맞지 않을 경우에 해당 테스트케이스에 대한 점수는 0점이다.

예제

Sample Input	Sample Output
3 4 PWGP WGPW GWPG	P-GP WGP G-PG

이 예제에서 3개의 아름다운 경단꼬치를 만들었다.

W G P의 순서로는 아름다운 경단꼬치를 만들 수 없음에 유의하여야.

Sample Input	Sample Output
3 4 PWWP WWW PGGP	PWWP W\W PGGP

이 예제에서 2개의 아름다운 경단꼬치를 만들었다.

시각화 도구

이 문제의 시각화 도구가 제공된다. 시각화 도구를 사용하면 입력 혹은 출력 파일의 개요를 볼 수 있다. 시각화 도구는 대회 홈페이지의 아카이브에서 다운받을 수 있다.

시각화 도구를 사용하려면, `visualizer.html`을 웹 브라우저로 열어서 파일을 선택하거나 드래그 앤 드롭해야한다. 시각화 도구가 주어진 형식을 완벽히 확인하지 않음에 유의하여라. 입력 파일의 형식이 틀리면 올바르게 동작하지 않을 수 있다. R 과 C 가 제한을 넘어가도 올바르게 동작하지 않을 수 있다.

문제 3. 치료 계획

입력 파일: standard input
출력 파일: standard output
시간 제한: 5 seconds
메모리 제한: 512 megabytes

JOI 왕국에는 1번 부터 N 번 까지 번호가 붙은 N 개의 집이 있다. 이 집은 번호순으로 일렬로 놓여있고, 각 집에는 한 명의 국민이 살고 있다. x 번 ($1 \leq x \leq N$) 집에 살고 있는 국민을 x 번 국민이라고 한다.

지금 신종 바이러스가 발생해서 모든 국민이 바이러스에 감염되었다. 이 문제를 해결하기 위해, M 개의 치료 계획이 제안되었다. i 번째 ($1 \leq i \leq M$) 치료 계획의 비용은 C_i 이다. i 번째 치료 계획이 실행 되면, 다음 일이 일어난다.

지금으로부터 T_i 일 후 저녁에, $L_i \leq x \leq R_i$ 를 만족하는 x 번 국민이 바이러스에 감염 된 경우, 그 국민은 치료된다.

바이러스는 다음과 같은 방식으로 인접한 국민들에게 퍼진다.

만약 x 번 ($1 \leq x \leq N$) 국민이 어느 날 아침 바이러스에 감염 되었다면, ($x \geq 2$ 인 경우) $x-1$ 번 국민과 ($x \leq N-1$ 인 경우) $x+1$ 번 국민은 같은 날 정오에 감염된다.

이미 치료된 국민도 다시 바이러스에 감염될 수 있다.

당신은 JOI 왕국의 총리이기 때문에, 다음 조건을 만족하도록 치료 계획 중 몇개를 골라야 한다.

조건 계획이 모두 실행 된 이후, 어떤 국민도 바이러스에 감염되어 있지 않다.

단, 같은 날에 두 개 이상의 계획을 실행하는 것도 가능하다.

집의 수와 치료 계획의 정보가 주어 졌을 때, 위 조건을 만족하면서 치료하는 것이 가능한지, 가능하다면 필요한 비용 합의 최솟값을 구하는 프로그램을 작성하여라.

입력 형식

표준 입력에서 다음과 같은 형식으로 주어진다. 모든 수는 정수이다.

N M

T_1 L_1 R_1 C_1

\vdots

T_M L_M R_M C_M

출력 형식

표준 출력에 하나의 줄을 출력하여라. 조건을 만족하지 못하는 경우 -1을 출력하여라. 조건을 만족할 수 있는 경우, 필요한 비용 합의 최솟값을 출력하여라.

제한

- $1 \leq N \leq 1\,000\,000\,000$.
- $1 \leq M \leq 100\,000$.
- $1 \leq T_i \leq 1\,000\,000\,000$ ($1 \leq i \leq M$).
- $1 \leq L_i \leq R_i \leq N$ ($1 \leq i \leq M$).

- $1 \leq C_i \leq 1\,000\,000\,000$ ($1 \leq i \leq M$).

서브태스크 1 (4 점)

- $T_i = 1$ ($1 \leq i \leq M$).

서브태스크 2 (5 점)

- $M \leq 16$.

서브태스크 3 (30 점)

- $M \leq 5000$.

서브태스크 4 (61 점)

추가 제한조건이 없다.

예제

standard input	standard output
10 5 2 5 10 3 1 1 6 5 5 2 8 3 7 6 10 4 4 1 3 1	7

이 예제에서, 다음과 같이 치료 계획을 진행 할 수 있다.

- 둘째 날 저녁, 1번 치료 계획 실행되어 5, 6, 7, 8, 9, 10번 국민이 치료되었다. 이제 1, 2, 3, 4번 국민은 바이러스에 감염되었다.
- 셋째 날 정오, 5번 국민이 바이러스에 감염되었다. 이제 1, 2, 3, 4, 5번 국민은 바이러스에 감염되었다.
- 넷째 날 정오, 6번 국민이 바이러스에 감염되었다. 이제 1, 2, 3, 4, 5, 6번 국민은 바이러스에 감염되었다.
- 넷째 날 저녁, 5번 치료 계획이 실행되어 1, 2, 3번 국민이 치료되었다. 이제 4, 5, 6번 국민은 바이러스에 감염되었다.
- 다섯째 날 정오, 7번 국민이 바이러스에 감염되었다. 이제 3, 4, 5, 6, 7번 국민은 바이러스에 감염되었다.
- 다섯째 날 저녁, 3번 치료 계획이 실행되어 3, 4, 5, 6, 7번 국민이 치료되었다. 이제 바이러스에 감염된 국민은 없다.

1, 3, 5번 치료 계획을 실행 하는 데에는 총 7의 비용이 든다. 조건을 만족하면서 7보다 더 작은 비용으로 치료 계획을 실행하는 것은 불가능하기 때문에, 7을 출력한다.

standard input	standard output
10 5 2 6 10 3 1 1 5 5 5 2 7 3 8 6 10 4 4 1 3 1	-1

이 예제에서, 조건을 만족시킬 수 없기 때문에 -1 을 출력한다.

standard input	standard output
10 5 1 5 10 4 1 1 6 5 1 4 8 3 1 6 10 3 1 1 3 1	7

이 예제는 서브태스크 1의 조건을 만족한다.