

회전 막대기 (rotate)

아사둘로는 세계적인 에너지 연구소 APIO에서 일하는 뛰어난 과학자이다. 최근 그는 알려지지 않은 물질로부터 에너지를 얻어내는 방법을 연구하고 있다.

이 물질은 그 자체가 에너지를 내지는 않지만, 이 물질로 만든 아주 가늘고 길다란 막대기 (두께가 없는 직선으로 간주할 수 있다) 여럿을 같이 모으면 상호 작용에 의해서 에너지를 만들어낼 수 있다.

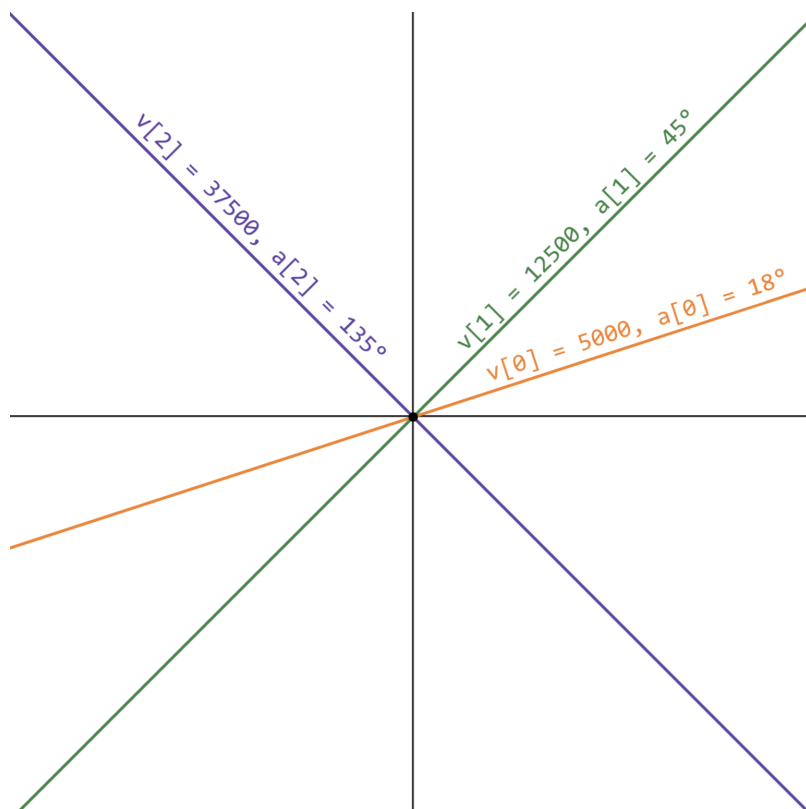
구체적으로는, n 개의 막대기에 대한 정보가 배열 $v[0], v[1], \dots, v[n-1]$ 에 주어진다. i 번째 막대기가 x 축의 양의 방향과 이루는 각은 반시계방향으로 $a[i] = 360 \cdot \frac{v[i]}{100000}^\circ$ 이다. n 개의 막대기가 만드는 에너지의 총량은

$$\sum_{i < j} \text{acute}(i, j)$$

여기서 $\text{acute}(i, j)$ 는 i 번째 막대기와 j 번째 막대기가 만드는 예각의 크기이다. 이 문제에서는 90° 도 예각으로 간주하자. 보다 엄밀하게 정의하면, $\text{acute}(i, j) = \min(|v[i] - v[j]|, 50000 - |v[i] - v[j]|)$ 이다.

다른 말로, 에너지의 총량은 가능한 모든 막대기의 쌍들이 이루는 예각의 크기의 합이다.

예를 들어, $v = [5000, 12500, 37500]$ 이고 이로부터 얻을 수 있는 각이 $a = [18^\circ, 45^\circ, 135^\circ]$ 이면, 다음과 같은 그림을 얻을 수 있다.



여기에서 $\text{acute}(0, 1) = 7500$ (즉 27°), $\text{acute}(0, 2) = 17500$ (즉 63°), $\text{acute}(1, 2) = 25000$ (즉 90°). 따라서, 이 막대기들로부터 얻을 수 있는 에너지의 총량은 $7500 + 17500 + 25000 = 50000$ 이다.

아사둘로는 n 개의 막대기를 잘 배치해서 최대의 에너지를 얻어내려고 한다. 그러나, 몇 가지 제약 조건이 있다.

- 먼저, 이 물질은 생명체에게 매우 위험하기 때문에, 특별한 기계 장치를 통해서만 막대기를 돌릴 수 있다. 이 장치는 여러 개의 막대기를 동시에 골라서 이 막대기들을 한번에 같은 각도만큼 돌릴 수 있다.
- 아사둘로는 에너지의 총량이 감소하게 하고 싶지 않다. 따라서, 기계 장치를 조작할 때 직전보다 에너지의 총량이 감소하게 만들면 안된다.
- 기계 장치를 조작하는데 많은 에너지를 소모하기 때문에, 모든 조작과정에서 고른 막대기 개수의 총합은 2 000 000을 넘어서는 안된다.

이런 제약조건을 가지고, 아사둘로는 막대기에서 얻어낼 수 있는 에너지 총량을 최대로 하도록 기계 장치를 조작하려고 한다. 아사둘로가 최대로 에너지를 얻어낼 수 있게 프로그램을 짜서 도와주자.

Implementation details

다음 함수를 구현해야 한다.

```
void energy(int n, std::vector<int> v)
```

- n : 막대기의 개수
- v : 막대기에 대한 정보를 저장하는 길이 n 인 배열
- 이 함수는 정확하게 한 번 호출된다.

이 함수 내부에서, 다음 함수를 호출할 수 있다.

```
void rotate(std::vector<int> t, int x)
```

- t : 서로 다른 인덱스를 저장한 배열. 즉, 각 i 에 대해서 $0 \leq t[i] < n$ 이고 $i < j$ 이면 $t[i] \neq t[j]$. 배열 t 는 꼭 정렬되지 않아도 된다.
- 이 함수는 인덱스가 t 에 포함된 모든 막대기를 동시에 x 만큼 회전시킨다. 즉, t 에 포함된 모든 인덱스 i 에 대해서 $v[i]$ 는 $(v[i] + x) \bmod 50000$ 가 된다.
- 이 함수는 여러번 호출될 수 있다. 모든 호출에서 t 의 길이의 총합은 2 000 000을 넘어서는 안된다.

Examples

Example 1

다음 호출을 생각해보자.

```
energy(2, [20000, 10000])
```

$v = [20000, 10000]$ 이고 에너지의 총량은 $20000 - 10000 = 10000$ 이다. 가능한 시나리오 중 하나는 다음과 같다.

- `rotate([0, 1], 8000)`를 호출한다. 그러면 v 는 $[28000, 18000]$ 가 된다. 에너지의 총량은 그대로이다.
- `rotate([0], 15000)`를 호출한다. 그러면 v 는 $[43000, 18000]$ 가 된다. 에너지의 총량은 $43000 - 18000 = 25000$ 이다.

주어진 입력에 대해서 최대로 얻을 수 있는 에너지의 총량이 25000라는 것을 보일 수 있다. 따라서, 아사들로는 더 이상 기계 장치를 조작할 필요가 없다.

Example 2

다음 호출을 생각해보자.

```
energy(3, [5000, 12500, 37500])
```

이 예제에 대한 그림이 위에 있다. 초기 상태에서 얻을 수 있는 에너지의 총량이 최대라는 것을 보일 수 있으며, 따라서 기계장치의 조작은 불필요하다.

Constraints

- $2 \leq n \leq 100\,000$
- 각 $0 \leq i < n$ 에 대해 $0 \leq v[i] \leq 49\,999$
- v 의 원소가 반드시 서로 달라야 하는 것은 아니다

Subtasks

1. (5 points) $n = 2$
2. (11 points) 각 $0 \leq i < n$ 에 대해 $v[i] < 25\,000$
3. (8 points) $n \leq 10$
4. (15 points) $n \leq 100$
5. (15 points) $n \leq 300$
6. (20 points) $n \leq 2000$
7. (26 points) 추가적인 제약 조건이 없다.

Sample Grader

샘플 그레이더는 다음 양식으로 입력을 읽는다.

- line 1: n
- line 2: $v[0] \ v[1] \ \dots \ v[n-1]$

샘플 그레이더는 다음 양식으로 출력한다.

- line 1: 최종적인 막대기들의 에너지의 총량

또한, 그레이더는 당신의 회전 조작에 대한 자세한 정보를 `log.txt` 파일에 출력한다.