Zadanie: WYC
Wycieczka

W Bajtocji jest n miast, niektóre z nich są połączone ze sobą dwukierunkowymi drogami. System drogowy Bajtocji jest tak zorganizowany, że między dowolną parą miast istnieje dokładnie jedna droga nieprzechodząca przez żadne miasto więcej niż raz.

Król Bajtocji chce odbyć wycieczkę rozpoczynającą i kończącą się w stolicy (mieście numer 1). W czasie tej wycieczki król chce odwiedzić każde miasto dokładnie raz (oprócz stolicy, którą król odwiedzi 2 razy). Doradcy króla szybko zauważyli, że bez stworzenia nowych dróg nie można zaplanować takiej podróży.

Twoim zadaniem jest odpowiedzieć na pytanie: ile trzeba wybudować nowych dróg, żeby król mógł odbyć wyżej opisaną wycieczkę ?

Wejście

W pierwszym wierszu znajduje się liczba naturalna n ($3 \leq n \leq 100\,000$), oznaczająca ilość miast w Bajtocji (numerowane są od $1 \ldots n$). W kolejnych $n-1$ wierszach znajdują się pary liczb $1 \leq a_i, b_i \leq n$. Każda taka para oznacza drogę między miastami a_i i b_i.

Wyjście

Powinno się składać z jednej liczby naturalnej k — minimalnej ilości dróg, które należy dobudować.

Przykład

Dla danych wejściowych:

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

poprawnym wynikiem jest:

```
2
```

Problem C. Good Students and Bad Students

Input file:	`standard input`
Output file:	`standard output`
Memory limit:	128 MB

$2kn$ students have entered the university this year, and we need to split them into n groups, each of size $2k$. Surprisingly, we've decided to take *their* opinion into account.

Each student has declared whether he wants to be in the stronger half of his group, or in the weaker half. More formally, each student is assigned a distinct integer *level*, and some students want to be in the first half of their group when sorted by level descending (mostly to feel good about themselves), and all other students want to be in the second half of their group in such ordering (to actually learn as much as possible from others).

You need to separate the students into groups in such a way that as many preferences are satisfied as possible.

Input

The first line of the input file contains two positive integers k and n ($1 \leq 2kn \leq 100\,000$). The second line contains $2kn$ distinct positive integers not exceeding 10^9, denoting the levels of the students. The third line contains $2kn$ integers, each either 1 if the corresponding student wants to be in the stronger half, or 0 if he wants to be in the weaker half.

Output

Output one integer — the number of preferences that we can satisfy.

Examples

standard input	standard output
3 2 15 14 12 11 8 7 6 5 4 3 2 1 1 1 0 1 0 1 0 0 0 1 1 0	8

Problem D. Grand Tour

Input file:	`standard input`
Output file:	`standard output`
Memory limit:	128 MB

Byteman, the head of Byteland parliament, has decided to take a grand tour and visit n leading universities of Byteland to talk to students about innovation and modernization. Visiting university takes exactly one day, so universities will be visited in some order on days $1, 2, \ldots, n$. But there are some limitations on the tour. Some universities are more important than others. More important universities must be visited earlier. Totally there are m constraints of a form "university x must be visited before university y".

Since it is difficult to gather students for a lecture during vacations, it is preferrable for Byteman to visit each university before summer vacations start there. For each university the last day of the semester d_i is known. If the last day of the semester in the university is d_i and Byteman visits it at day a_i, *discomfort* for this university is $\max(0, a_i - d_i)$. The *badness* of the tour is maximal discomfort of a university caused by it.

Officials want to plan the tour in such way that badness of the tour was as small as possible.

Input

Input contains several test cases.

The first line of each test case contains n — the number of universities ($1 \le n \le 100\,000$). The second line contains n integers: $d_1, d_2, \ldots, d_n$ — the last days of semester in each university ($1 \le d_i \le 10^9$). The following line contains m — the number of constraints ($0 \le m \le 100\,000$). The following m lines contain two integers each, pair x, y means that university x must be visited before university y. Universities are numbered from 1 to n. The set of constraints is consistent.

The last test case is followed by $n = 0$ which must not be processed. Both the sum of n and the sum of m in each input file doesn't exceed 10^5.

Output

The first and only line must contain the minimal possible badness of the tour.

Examples

standard input	standard output
6 4 5 6 3 5 3 7 1 2 2 3 4 3 5 2 5 4 6 5 6 1 0	0