B. Quadrivial (Quadrivial)

Problem Statement

Kitty enjoys competitive programming, but she is currently troubled by setting new problems for NHSPC.

"How on earth do I compose a non-trivial problem?" Day after day she wondered.

For this, Kitty scoured through theses, studied new tricks, discussed algorithms with others (like quad-connected component in $O(n+m)$). Even after all that, she still couldn't compose a non-trivial problem.

In the late, dark night, a day before the deadline, with Laffey by her side, an idea struck: Why not compose a problem about problem composing?

Kitty and Laffey worked together to write down every problem they knew and every trick they studied in the form of integer sequences. In a **brainstorming session**, they permute all problems and tricks and chains them back to back to form a long sequence, after that, in a **composing session**, they pick a non-empty interval from that sequence to be their problem. The greatness of the problem is the sum of the integers in that interval. Kitty is interested in the greatest problem possible from all possible brainstorming and composing sessions.

More formally, let's say Kitty and Laffey knew n problems and tricks, the i-th of them can be written as an integer sequence $a_{i,1}, a_{i,2}, \ldots, a_{i,k_i}$. Kitty first decides a permutation of 1 to n $p_1, p_2, \ldots, p_N$, and connects the sequences p_1-th problem, p_2-th problem, ..., p_n-th problem in that order. Kitty then chooses a non-empty interval from the connected sequences and computes the sum of the integers in that interval. She is interested to know the largest possible sum that can be obtained.

Smart as Kitty is, she noticed instantly that connecting problems and tricks doesn't make a problem non-trivial, instead, it's even more trivial! She didn't have time to compose another problem so this is the one she submitted. As the problem is more trivial than ever, she decides to call it "Quadrivial".

Input Format

$$
\begin{array}{l}
n \\
k_1 \ a_{1,1} \ a_{1,2} \ \cdots \ a_{1,k_1} \\
k_2 \ a_{2,1} \ a_{2,2} \ \cdots \ a_{2,k_2} \\
\vdots \\
k_n \ a_{n,1} \ a_{n,2} \ \cdots \ a_{n,k_n}
\end{array}
$$

- n denoted the number of integer sequences.
- k_i denotes the size of the i-th sequence.
- $a_{i,j}$ denotes the j-th element in the i-th sequence($1 \le i \le n$, $1 \le j \le k_i$).

Output Format

ans

- ans denotes the greatness of greatest problem possible.

Constraints

- $1 \leq n \leq 10^5$
- $1 \leq k_i \leq 10^5 (1 \leq i \leq n)$
- The sum of lengths of the n sequences, $\sum_{i=1}^{n} k_i \leq 10^6$
- $|a_{i,j}| \leq 10^9 (1 \leq i \leq n \text{、} 1 \leq j \leq k_i)$
- All inputs are integers

Example

Sample Input	Sample Output
4 3 1 4 -6 2 -1 -3 6 6 -11 5 -13 3 12 4 14 -7 -3 15	39
1 1 -1000000000	-1000000000

Scoring

There are 4 subtasks in this problem. The score and additional constraints of each subtask are as follows:

Subtask	Score	Additional constraints
1	8	$n = 1$
2	14	$n \leq 400$
3	22	$n \leq 3000$
4	56	No other constraints