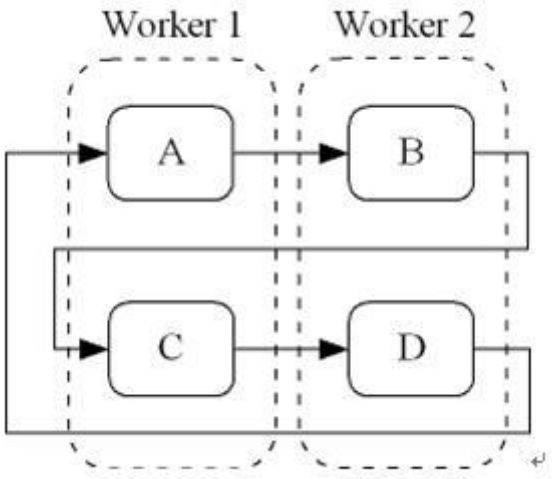
Problem 1. Re-Entrant System Simulation Problem

(Time Limit: 2 seconds)

Problem Description

Suppose there are two workers and four machines A, B, C, and D. The first (respectively, second) worker needs to operate machine A and C (respectively, B and D). In addition, the priority of A (respectively, B) is higher than C (respectively, D), i.e. if both A and C (respectively, B and D) have products need to operate, then the worker will first operate A (respectively, B). Note that if the worker is operating a machine, then the worker will not interrupt operating current machine. Suppose there are 4 wafers and the progress of manufacturing is A -> B -> C -> D -> A -> B -> C -> D -> A -> … . We call four operations on the first wafer from A to D as one round. Each machine has a queuing buffer and will operate the first coming wafer. Suppose the operating time of each machine is one unit time and all wafers are in machine A at the beginning. Please write a program to simulate such operation environment for the input round and finally print how much unit time it will spend.



Technical Specification

1. There are two kind of ideas for this simulation, one is to record the statements of each wafer, the other is to record the statement of each machine and only the first wafer. We suggest the second idea.
2. If you are going to use the second idea, you should keep some information of the first wafer, includes in which machine and how many wafers are in front of it.

3. You should be attention the synchronization. That is, if both of workers are done in the same time, you should first update the statements of all machines and then choose which machine to process.
4. There are two workers, four machines, and four wafers.

Input Format

The first line of the input contains an integer indicating the number of test cases. For each test case, there contains a line with an integer n ($1 \leq n \leq 100000$) indicating the number of rounds.

Output Format

For each test case, print the total time units during which the first wafer finishes the input rounds of operations in each line.

Example

Sample Input:	Sample Output:
3 1 2 10	6 14 78

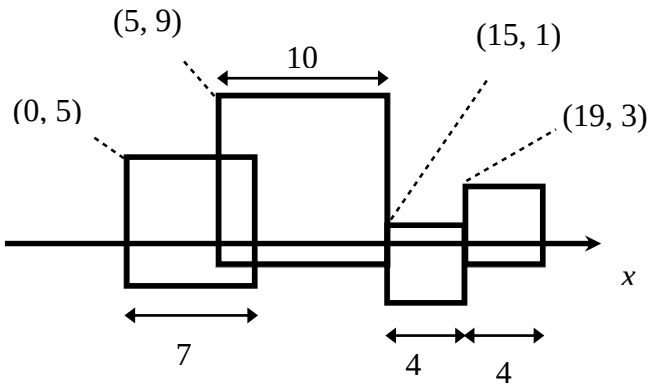
Problem 2. Room Decoration

(Time Limit: 1 second)

Problem Description

A designer is going to decorate a room with a "plate wall." On the wall there is a wood stick, and a number of square-shaped paper plates of distinct sizes are to be fixed on the wall. Each square plate is fixed so that two edges are parallel to the stick. A plate can be fixed by nailing somewhere inside it to the wood stick. After deciding the positions of the plates, the designer asks you to help her find the minimum number of nails with which she can fix the plates on the wall.

Conceptually, the wood stick is assumed to be the x-axis and plates are squares. Multiple plates can be fixed by one nail if their intersection is nonempty. Notice that you nail a plate to the wood stick by hammering a nail somewhere "inside" the plate, excluding the edges. For example, in the following figure, we need three nails to fix the plates.



Technical Specification

- The number of plates to be fixed is denoted by n, which is a positive integer at most 20000.
- There are n positive integers $l_1, l_2, \ldots, l_n$, indicating the lengths of edges of the plates. For $1 \le i \le n$, $2 \le l_i \le 6300$.
- There are n coordinates $(x_1, y_1), \ldots, (x_n, y_n)$, which are the coordinates of the top-left corners of the plates, where x_i and y_i are integers. For $1 \le i \le n$, $0 \le x_i \le 20000$ and $1 \le y_i \le 6300$. Each square intersects the x-axis.

Input Format

The first line of the input contains an integer indicating the number of test cases. Each test case contains n+1 lines. The first line is the number n. Each of the following n lines consists of three integers. The first integer is the edge length, followed by the x- and y-coordinates. Consecutive integers are separated by a space.

Output Format

For each test case, please output the minimum number of nails needed to fix the plates in each line.

Example

Sample Input:	Sample Output:
2 3 2 1 1 4 3 2 2 7 1 4 3 1 2 2 5 1 5 3 3 5 4 1	3 2

Problem 3. Hotel Booking for a Tour Group

(Time Limit: 3 seconds)

Problem Description

Hotel Five is accepting a reservation request from a tour group. There are N rooms available for a group of size M such that $M \leq N$. The rooms in the hotel are divided into five categories (denoted by A, B, C, D, E). Based on the preferences of guests in the group, for each individual guest, only two types of rooms meet his/her requirement. You must write a program to decide if the hotel manager can assign rooms in such a way that all guests' requirements can be satisfied.

Input Format

The first line of the input contains the number of test cases t, $1 \leq t \leq 50$. For each test case, the first line contains five numbers a, b, c, d, e that indicate the number of available rooms of types A, B, C, D, E respectively, where $0 \leq a, b, c, d, e \leq 70$. The second line contains a number M indicating the number of guests in the group with $1 \leq M \leq N = a + b + c + d + e$. Subsequently, M lines are listed where each line contains, separated by one space, the two types of rooms that satisfy the requirement of an individual guest.

Output Format

For each test case, print a line containing '`YES`' if there is, at least, a way of room assignment that can meet all guests' requirement. Otherwise, print '`NO`'

Example

Sample Input	Sample Output
3 3 5 0 2 8 6 B C B C A B E F D E	YES YES NO

C D 1 2 1 1 1 4 B E C E B C B C 2 3 0 0 1 1 C D	

Problem 4. Alignment with Affine Penalties

(Time Limit: 1 second)

Problem Description

In bioinformatics, a sequence alignment is a way of arranging the sequences of DNA, RNA, or protein in order to measure the similarity that may be a consequence of evolutionary mutations between two sequences. Each character in the sequence will be aligned (i.e., matched/mismatched) to the character on the other sequence or aligned to a gap symbol. Gaps are inserted into the sequences so that identical or similar characters may be aligned in successive columns. Given a scoring function, match=+1, mismatch=-2, and gap=-1, the alignment with highest score is considered as the most-likely evolutionary change between the two sequences. In reality, the number of gaps in an alignment tend will be reduced and grouped together, which makes more biological sense as a large insertion or deletion is more likely than multiple small and interspersed insertions/deletions. For instance, the following left alignment is preferred over the right one, because only one large deletion instead of two smaller deletions occurred.

```
TATCCCCGAG          TATCCCCGAG
TATCC--GAG          TATC-C-GAG
```

Consequently, an extension to linear gap costs, called affine gap penalty, for preferring the grouped gaps is the usage of two different gap penalties: gap opening penalty (e.g., -4) and original gap extension penalty (e.g., -1). That is, for each region with consecutive gaps, additional gap opening penalty is charged in conjunction with the original gap extension penalties. For example, the above left alignment has score 2 (i.e., 8 matches, 1 gap opening penalty, and 2 gap extension penalty) while the right one has score -2 (i.e., 8 matches, 2 gap open penalties, and 2 gap extension penalties). Therefore, the left alignment is preferred over the right one according to the score. Given a scoring function and two sequences, you are asked to compute the maximum score of alignment considering the gap open penalty.

Technical Specification

- The length of each sequence ranges from 3 to 100.
- The scoring function is match: +1, mismatch: -2, gap open: -4, gap extension: -1.

Input Format

The first line of the input file contains an integer indicating the number of test cases. Each test case consists of two lines, where the first and second line stores the first and second sequences.

Output Format

For each test case, output the maximum alignment score in each line.

Example

Sample Input:	Sample Output:
2 ATA ACA TATCCCCGAG TATCCGAG	0 2

Problem 5. Maximum Squares

(Time Limit: 10 seconds)

Problem Description

An H-V polygon is a polygon consisting of only horizontal or vertical line segments. Similarly, an H-V square is a square that satisfies the conditions of an H-V polygon. In this problem, given an H-V polygon, we want to find the largest H-V square that can fit inside the polygon.

For example, in Figure 1, the largest H-V square that can fit inside the polygon is of size 4×4.

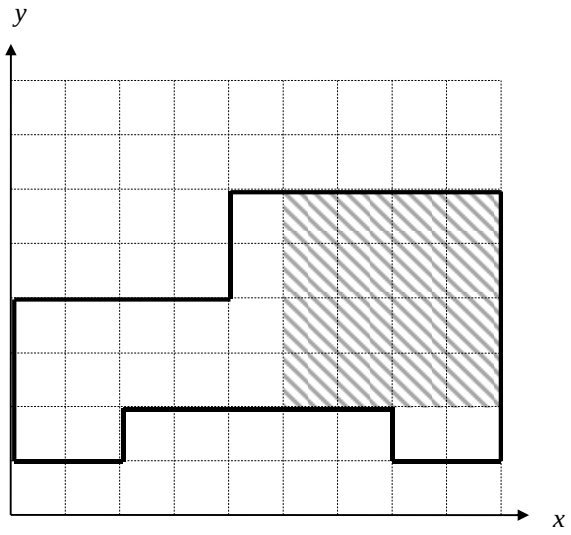


Figure 1.

Given an H-V polygon *P*, please find the largest H-V square that can fit inside *P*.

Technical Specification

- The number *n* of vertices of *P* is a positive integer between 4 and 1000.
- The coordinates *x*, *y* of each vertex are nonnegative integers between 0 and 10000.
- The given polygon *P* is simple, which means that the boundary of *P* does not cross itself

Input Format

The first line is an integer t, $1 \le t \le 10$, indicating the number of test cases. Each test case starts with one line with a positive integer n, $4 \le n \le 1000$, indicating the number of vertices of the polygon P. Then, n lines follow, each of which contains 2 integers x, y, $0 \le x, y \le 10000$, indicating the coordinate of a vertex. The vertices are given in counterclockwise order.

Output Format

For each test case, output the area of the largest H-V square that can fit inside P in each line.

Example

Sample Input:	Sample Output:
2 4 0 0 5 0 5 7 0 7 10 0 1 2 1 2 2 7 2 7 1 9 1 9 6 4 6 4 4 0 4	25 16