Problem A: Luanda

(blue balloon)

Luanda municipality has set up a new charging method for the Congestion Charging Zone (CCZ) which controls the passage of vehicles in Luanda's high-congestion areas in the congestion period (CP) from 6:30 to 19:00. There are plate detection cameras inside or at the entrances of the CCZ recording vehicles seen at the CCZ. The table below summarizes the new charging method.

The first time seen in the CP	The last time seen in the CP	Charge
6:30 to 10:00	6:30 to 16:00	24000
6:30 to 10:00	16:01 to 19:00	36000
10:01 to 16:00	10:01 to 16:00	16800
10:01 to 19:00	16:01 to 19:00	24000

Note that the first time and the last time that a vehicle is seen in the CP may be the same. Write a program to compute the amount of charge of a given vehicle in a specific day.

Standard Input

The first line of the input contains a positive integer n ($1 \leqslant n \leqslant 100$) where n is the number of records for a vehicle. Each of the next n lines contains a time at which the vehicle is seen. Each time is of form <hour>:<minute>, where <hour> is an integer number between 0 and 23 (inclusive) and <minute> is formatted as an exactly two-digit number between 00 and 59 (inclusive).

Standard Output

Print the charge to be paid by the owner of the vehicle in the output.

Examples

Sample Input	Sample Output
4 7:30 2:20 7:30 17:30	36000

Sample Input	Sample Output
1 12:13	16800

Sample Input	Sample Output
2 0:30 23:30	0

Problem B: ACPC

(red balloon)

The African ChamPions Cup (ACPC), the most prestigious football league in Africa, is reaching its end, and people are eagerly waiting for the finals, which happened to be between the two most popular African teams, Zamalek and Mazembe.

The ACPC finals consist of two matches, with each team competing as the home team in one match. The winning team is determined by aggregate score, the sum of the scores of the two matches. For example, if the scores of the two matches are Zamalek 6–0 Mazembe in the first match, and Mazembe 3–1 Zamalek in the second match, then the aggregate score will be Zamalek 7–3 Mazembe, meaning that Zamalek is the winner. If aggregates are equal, the away goals rule is used to determine the winner, in which case the winner is the team that scored the most goals in the match it played away from home. If the result is still equal, a penalty shootout is required.

Augusto, an avid football fan, is trying to figure out various scenarios in which her favorite team wins the finals. To this end, he aims to write a program that gets as input the number of goals in the two matches, and decides which team is the winner if it can be derived from the aggregate scores and the away goals rule, otherwise declares that the match goes to penalty kicks. You are going to help Augusto write such a program.

Standard Input

The first line of the input contains two space-separated integers p_1 and s_1, where p_1 and s_1 are the number of goals scored by Zamalek and Mazembe, respectively, in the first match in which Zamalek is the home team. The second line contains two space-separated integers s_2 and p_2, where s_2 and p_2 are the number of goals scored by Mazembe and Zamalek, respectively, in the second match in which Mazembe is the home team. All input integers are between 0 and 20, inclusively.

Standard Output

In the output, print the name of the winning team, either `Zamalek` or `Mazembe`, if the winner can be determined by the aggregate scores and the away goals rule. Otherwise, print `Penalty`.

Examples

Sample Input	Sample Output
3 0 2 1	Zamalek

Sample Input	Sample Output
3 1 2 0	Mazembe

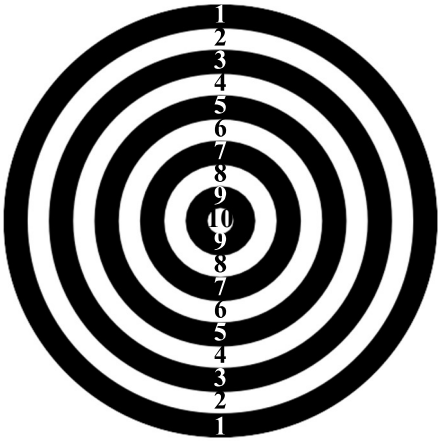
Sample Input	Sample Output
2 0 2 0	Penalty

Problem C: MarathonChallenge

(yellow balloon)

MarathonChallenge is a gathering event at Mapcom similar to TGIF events at Google. Some entertainment programs like pantomime, foosball, Xbox/PS4, and several board games are part of the event. You are going to set up a dart game in MarathonChallenge. As a techie organizing a game for techies, you would rather use a smart screen and write a program to calculate the scores instead of hanging a traditional dartboard and scoring the shots manually. Your program must get the coordinates of dart shots for a player and calculate his/her total score. The score for each dart shot (at point (x, y)) is calculated based on its distance from the center of the dartboard (point $(0, 0)$). If the distance is d millimeters, the score is calculated based on the following table:

Condition	Score
$d \leqslant 10$	10
$10 < d \leqslant 30$	9
$30 < d \leqslant 50$	8
$50 < d \leqslant 70$	7
$70 < d \leqslant 90$	6
$90 < d \leqslant 110$	5
$110 < d \leqslant 130$	4
$130 < d \leqslant 150$	3
$150 < d \leqslant 170$	2
$170 < d \leqslant 190$	1
$190 < d$	0



Standard Input

The first line of the input contains a single integer N as the number of dart shots for a player ($1 \leqslant N \leqslant 100$). Each of the next N lines contains two space-separated integers as the coordinates (x, y) of a dart shot. The coordinates are in millimeters and their absolute values will not be greater than 300.

Standard Output

Print a single line containing the total score of the player.

Examples

Sample Input	Sample Output
2 4 7 -31 -5	18

Sample Input	Sample Output
3 12 -16 -180 100 152 10	11

Problem D: Movie
(Orange balloon)

The main movie theater of the city consists of a single auditorium with rows of comfortable padded seats. Surprisingly, the comfortness of seats are not necessarily equal. Precisely, each seat has its own comfort value which is a non-negative integer number. A seat is more comfortable than another seat if its comfort value is larger. It is only possible to enter a row from the left side of the auditorium. Assume the seats in a row are numbered 1 to m from left to right. When a person enters a row, he/she always sits on the most comfortable seat which is free and accessible to him/her. If he/she sits at seat i, he/she blocks other persons coming later to sit on seats $i+1$ to m. If there are more than one free and accessible seat being the most comfortable, he/she sits on the leftmost one. The owner of the movie theater plans to improve the comfortness of some seats to have more audiences in the auditorium. Improving one unit in the comfortness of a seat costs some fix value. With the budget available, the owner knows the total improvement over all seats must not exceed a value k. Help the owner find the best way to improve the comfortness of seats by at most k units in total to have the maximum number of audiences in the auditorium.

Standard Input

The first line contains three non-negative integers n, m, and k which are the number of rows, the number of seats in each row, and the total comfortness that can be added to all seats. The next n lines describe the comfort values of seats; each line contains m non-negative integers not more than 10^6 denoting the comfort values of seats from left to right for a row.

$(1 \leqslant n \cdot m \leqslant 3 \times 10^5, 0 \leqslant k \leqslant 10^{12})$

Standard Output

Print a single line containing the maximum number of audiences.

Examples

Sample Input	Sample Output
2 3 10 10 1 12 8 3 6	5

Sample Input	Sample Output
1 4 6 9 8 10 8	3

Sample Input	Sample Output
1 3 2 10 10 10	2

Problem E : Telco

(purple balloon)

Telco which is a new telecommunications company, plans to install its base stations in the city. The places where base stations must be installed have been already specified. Telco has two types of antennas to be used in the base stations: (i) antennas with transmission radius a, and (ii) antennas with transmission radius b. Two antennas can communicate with each other if and only if both are inside the coverage area of each other. Antenna with smaller transmission radius of course is cheaper. Telco plans to minimize its cost while keeping the whole network connected. Precisely, Telco plans to minimize its cost which is the sum of the transmission radii of all antennas. Interestingly, all base-station places are on a line. Help Telco construct a connected network with the minimum cost.

Standard Input

The first line of the input contains three positive integers n, a and b ($1 \leqslant n \leqslant 10^5$ and $1 \leqslant a, b \leqslant 10^5$) where n is the number of base stations, and a and b are radii as defined in the problem statement. The second line contains n distinct coordinates of base stations on the line with respect to an origin on the line. All coordinates are positive integers not more than 10^5.

Standard Output

If it is possible to construct a connected network, print the minimum cost in the output. Otherwise, print `-1` .

Sample Input	Sample Output
3 1 3 1 4 3	7