

## Rummy (rummy)

There are  $N + 2$  friends, indexed from 0 to  $N + 1$ , who gathered to play a game of Rummy.



Figure 1: The group of friends playing Rummy.


After playing many rounds, each friend obtained some score. The score of friend  $i$  is an integer  $P_i$  (in Rummy, scores can be negative). Friends 0 and  $N + 1$  obtained a non-negative score. This made the other  $N$  friends jealous, so they devised the following *maneuver* in order to make every score non-negative:

- Friend  $i$  ( $1 \leq i \leq N$ ) gives one point to each of they two neighbours.

In other words, after friend  $i$  applies the maneuver, their score decreases by two ( $P_i = P_i - 2$ ), and the scores of friends  $i - 1$  and  $i + 1$  increase by one ( $P_{i-1} = P_{i-1} + 1$ ,  $P_{i+1} = P_{i+1} + 1$ ). Note that friends 0 and  $N + 1$  are not allowed to apply the maneuver.

Each friend can apply the maneuver as many times as they want, even if their score is negative, or becomes negative after the maneuver.

Determine whether every friend can have a non-negative score (all at the same time) after applying some number of maneuvers.

 Among the attachments of this task you may find a template file `rummy.*` with a sample incomplete implementation.

## Input

The first line of the input file contains a single integer  $T$ , the number of test cases.  $T$  test cases follow, each preceded by an empty line.

Each test case consists of:

- a line containing the integer  $N$ ;
- a line containing the  $N + 2$  integers  $P_0, \dots, P_{N+1}$ .

## Output






For each test, case print "YES" if there is a sequence of 0 or more maneuvers such that in the end each friend will have a non-negative score, otherwise print "NO".

## Constraints

- $1 \leq T \leq 200\,000$ .
- $2 \leq N \leq 200\,000$ .
- $-1\,000\,000\,000 \leq P_i \leq 1\,000\,000\,000$  for each  $i = 0, 1, \dots, N + 1$ .
- $P_0, P_{N+1} \geq 0$ .
- The sum of  $N$  across all testcases does not exceed 200 000.

## Scoring

Your program will be tested against several test cases grouped in subtasks. In order to obtain the score of a subtask, your program needs to correctly solve all of its test cases.

- Subtask 1 (0 points)      Examples.  

- Subtask 2 (6 points)       $N = 2$ .  

- Subtask 3 (22 points)       $N = 3$ .  

- Subtask 4 (38 points)       $T \leq 20, N \leq 1000$ .  

- Subtask 5 (34 points)      No additional constraints.  


## Examples

input	output
5	YES
2	YES
12 -10 20 0	NO
3	YES
7 -8 19 -10 5	NO
10	
4 40 0 0 0 0 0 0 0 0 -6 0	
10	
4 40 0 0 0 0 0 0 0 0 -4 0	
4	
6 -10 7 8 -12 25	

## Explanation

In the **first test case** of the **sample case**, friend 2 can apply the maneuver 10 times. In the end, the scores will be as follows: 12, 0, 0, 10.

In the **second test case** of the **sample case**, friend 2 can apply the maneuver 10 times. The scores will become: 7, 2, -1, 0, 5.

Then, friend 1 will apply the maneuver once, resulting in the following scores: 8, 0, 0, 0, 5.