Algorithm Design Laboratory with Applications

Prof. Stefano Leucci

Problem: *Chess boxing.*

Bruto Arroccatore is a contender for the world title in *Chess Boxing*, a sport that alternates boxing matches with chess games. Overall, there are $n \geq 3$ athletes registered for the world tournament and they indexed with integers from 1 to n, and Bruto has index n. The tournament is point-based and consists of a series of matches in which two athletes compete against each other. A generic match between the i-th athlete and the j-th athlete, with $1 \leq i < j \leq n$, can thus be described by the pair (i, j). Each match end with either a winner or in a draw, and distributes a total of 2 points to the contestants. In particular, the winner of the generic match (i, j), if it exists, receives 2 points (and the loser receives no points). If the match ends in a draw, both participants receive 1 point.

We are in the advanced stages of the tournament, there are $m \geq n$ remaining matches, and the generic i-th athlete earned p_i points so far.

Design an efficient algorithm that receives n, m, the points $p_1, \ldots, p_n$, and the list of remaining matches as input, and decides if it is possible for Bruto place first at the end of the tournament, with no other athlete tied for the first place. In other words, the algorithm must decide if there exists a combination of results for the remaining m matches ensuring that the overall number of points earned by Bruto throughout the tournament is strictly larger that those earned by any other athlete.

Input. The input consists of a set of instances, or *test-cases*, of the above problem. The first line contains the number T of test-cases. The first line of each test-case contains the integers n and m. The second line of the test-case contains the n integers $p_1, p_2, \ldots, p_n$. Finally, the k-th of the subsequent m lines describes the k-th match (i, j) among the m remaining matches and contains the two integers i and j.

Output. The output consists of T lines. The i-th line is the answer to the i-th test-case, that is, the character "`Y`" if it is possible for Bruto to win the tournament (with an untied first place), and the character "`N`" otherwise.

Assumptions. $1 \leq T \leq 10$; $3 \leq n \leq 2^9$; $n \leq m \leq 2^{13}$; $\forall i = 1, \ldots, n,\ p_i \leq 2^{12}$.

Example. If $n = 5$, $m = 6$, $p_1 = 11$, $p_2 = 8$, $p_3 = 10$, $p_4 = 10$, $p_5 = 9$, and the remaining matches are $(1, 2)$, $(1, 3)$, $(2, 5)$, $(1, 4)$, $(3, 5)$, $(2, 3)$, one possible combination of results that allows Bruto to remain at the top of the standings at the end of the tournament is as follows: the matches $(1, 2)$ and $(2, 3)$ are won by 2, the match $(1, 4)$ is won by 4, the matches $(2, 5)$ and $(3, 5)$ are won by 5, and the match $(3, 4)$ ends in a draw. In this way, at the end of the tournament athlete 1 has $11 + 1 = 12$ points; athlete 2 has $8 + 4 = 12$ points; athlete 3 has $10 + 1 = 11$ points; athlete 4 has $10 + 2 = 12$ points, and Bruto (athlete 5) has $9 + 4 = 13$ points.

(Continues on the next page)

Input (the first test-case corresponds to the example above):

```
2
5 6
11 8 10 10 9
1 2
1 3
2 5
1 4
3 5
2 3
5 6
11 9 11 11 9
1 2
1 3
2 5
1 4
3 5
2 3
```

Output:

```
Y
N
```

Requirements. Your algorithm should require $O(m^2)$ time (with reasonable hidden constants).
Notes. A reasonable implementation should not require more than 5 seconds for each input file.