**Take discount or Not**

There are N*N* items in a shop. You know that the price of the i*i*-th item is Ai*Ai*​. Chef wants to buy all the N*N* items.

There is also a discount coupon that costs X*X* rupees and reduces the cost of every item by Y*Y* rupees. If the price of an item was initially ≤Y≤*Y*, it becomes free, i.e, costs 00.

Determine whether Chef should buy the discount coupon or not. Chef will buy the discount coupon if and only if the total price he pays after buying the discount coupon is **strictly less** than the price he pays without buying the discount coupon.

**Input Format**

* The first line of input will contain a single integer T*T*, denoting the number of test cases. The description of the test cases follows.
* Each test case consists of two lines of input.
  + The first line of the test case contains three space-separated integers — N*N*, X*X*, and Y*Y*.
  + The second line contains N*N* space-separated integers — A1,A2,…,AN*A*1​,*A*2​,…,*AN*​.

**Output Format**

* For each test case, output COUPON if Chef should buy the discount coupon, and NO COUPON otherwise.

Each letter of the output may be printed in either lowercase or uppercase. For example, the strings coupon, CouPoN, and COUPON will all be treated as equivalent.

**Constraints**

* 1≤T≤10001≤*T*≤1000
* 1≤N≤1001≤*N*≤100
* 1≤X,Y≤1051≤*X*,*Y*≤105
* 1≤Ai≤1051≤*Ai*​≤105
* **Sample 1:**
* Input
* Output
* 5
* 4 30 10
* 15 8 22 6
* 4 40 10
* 15 8 22 6
* 4 34 10
* 15 8 22 6
* 2 10 100
* 60 80
* 3 30 5
* 50 60 50
* COUPON
* NO COUPON
* NO COUPON
* COUPON
* NO COUPON