Bring Down the ~~Sky~~ Grading Server (grading-server)

Despite running low on oil, the opening ceremony was a great success.* Now, the committee is looking forward to the first competition day. But what's this? The chairman of the Technical Committee noticed some suspicious network activity—apparently someone is planning to hack the grading server!

The grading server possesses a specific computing power c_G which the mysterious hacker will try to reduce to (or below) 0. But the server is also protected by f_G firewalls, each of which will reduce the impact of any attack by a fixed amount S. At each point in time, the hacker can now decide to either

 ▶ take down one of these firewalls, permanently reducing f_G by 1 (down to a minimum of 0), *or*
 ▶ use all of their own computing power c_H to attack the server, permanently reducing c_G by $\max\{c_H - f_G \cdot S, 0\}$.

However, the chairman can strike back by either taking down one of the hacker's f_H firewalls or by using the grading server's computing power to attack the hacker, which will similarly reduce c_H by $\max\{c_G - f_H \cdot S, 0\}$. The chairman and the hacker take turns in their actions, with the hacker going first.

The committee will not know how much computing power and how many firewalls the hacker possesses until the hack has started. At the same time, because the committee might still be able to upgrade the server, its computing power and the number of its firewalls are also still unknown. To plan their defense, the committee therefore needs a program that tells them for Q different scenarios (c_H, f_H, c_G, f_G) whether the hacker can bring down the grading server even if the chairman makes optimal decisions.

Input

The first line of the input contains the integers S and Q. Then Q lines follow, each of which describes a scenario as above and consists of four integers c_H f_H c_G f_G: the computing power and number of firewalls of the mysterious hacker and of the grading server, respectively.

Output

Your program should output Q lines, each of which should consist of a single string: YES if the hacker can reduce the grading server's computing power to or below 0 in the corresponding scenario (regardless of the chairman's actions), and NO otherwise.

Constraints

We always have $1 \le S \le 30\,000$, $1 \le c_H, c_G \le 10^{12}$, $0 \le f_H, f_G \le 10^{12}$, and $1 \le Q \le 250\,000$.

Subtask 1 (5 points). $S, c_H, f_H, c_G, f_G \le 75$

Subtask 2 (5 points). $S, c_H, f_H, c_G, f_G \le 300$

Subtask 3 (10 points). $S = 1$

Subtask 4 (25 points). $S, c_H, f_H, c_G, f_G \le 2\,000$

Subtask 5 (20 points). $S \le 400$

Subtask 6 (20 points). $f_G, f_H \le 125$

Subtask 7 (15 points). No further constraints.

---

* At least if one ignores the grand finale with the stage collapsing under the weight of 10^{17} performers.

Examples

Input	Output
17 2 42 1 33 1 42 1 33 7	YES NO
1 1 999999999999 999999999999 999999999999 999999999999	YES
2 1 1000000000000 0 1 1000000000000	NO

Consider the first scenario of the first example:

▶ In the beginning, the hacker can attack the grading server, reducing c_G by $42 - 1 \cdot 17 = 25$ to a new value of 8.

▶ Afterwards, the chairman cannot reduce the hacker's computing power by attacking, so the only sensible thing for him to do is to take down the hacker's unique firewall.

▶ However, the hacker can then reduce the computing power of the grading server to $8 - 25 = -17 \leq 0$ by another attack, bringing down the server and ruining the first competition day.

In the second scenario on the other hand:

▶ Initially, the only thing the hacker can do is to take down one of the firewalls.

▶ Afterwards, the chairman can attack the hacker, reducing their computing power c_H to 26.

▶ In the next two rounds, the hacker again can only take down one of the firewalls, while the chairman can attack each time, thus reducing c_H below 0 and successfully fighting off the hacker's attempt.

Limits

Time: 4 s
Memory: 1024 MiB