

Прямокутники всюди

1. Відповідь «NO» — тоді і тільки тоді, коли $(x_1 = 0 \text{ і } x_2 = n)$ або $(y_1 = 0 \text{ і } y_2 = m)$.
2. Відповідь «NO» — тоді, коли або один з прямокутників виконує умову з блока 1, або один з прямокутників торкається лівої і нижньої пари стінок, інший — верхньої і правої і ці два прямокутники перетинаються
3. Потрібно перевіряти для кожної пари умову блока 2, також якщо прямокутники не перетинаються, але обидва перетинаються з третім — то відповідь теж — «NO»
4. Можна побудувати граф між всіма точками та лініями перетину та запустити BFS з пари сторін (ліво, низ) і перевірити, чи він дійшов до пари сторін (право, верх), або стиснути координати і розв'язувати задачу на матриці $(2k, 2k)$ буквально для кожної клітинки перевіряючи чи можна в неї зайти.
5. Можна як і в блоці 4 побудувати матрицю і завдяки двовимірним префікс-сумам дізнатись, які клітинки доступні. Далі все як в блоці 4.
6. Дістатись з точки $(0, 0)$ в точку (n, m) не можливо тоді і тільки тоді, коли між верхньою і правою парою стінок кімнати, та лівою і нижньою парою можливо пройти лише по прямокутниках. Будемо перевіряти це наступним чином: додамо як стартові точки BFS в чергу усі прямокутники які дотикаються однієї пари стінок і за $O(n^2)$ перевіримо чи можливо дійти до іншої пари стінок, фактично ми перевірятьмо чи існує послідовність прямокутників така, що шлях між парами стінок проходить через ці прямокутники і лише через них, якщо два прямокутники мають спільну точку, то з будь-якої точки одного дійти до будь-якої точки іншого. Прошу помітити, що тримати матрицю суміжностей між прямокутниками не обов'язково, можна для кожного прямокутника коли в BFS настає його черга - перевіряти усі інші прямокутники на те, чи мають вони спільну точку.

Тут можна почитати про алгоритм bfs: <https://algoua.com/algorithms/graphs/bfs/>

Тут можна почитати про двовимірні префікс суми: <https://usaco.guide/silver/more-prefix-sums?lang=cpp>