C. ETA

Ограничение времени	1 секунда
Ограничение памяти	64 Мб
Ввод	стандартный ввод или input.txt
Вывод	стандартный вывод или output.txt

При вызове такси мы не всегда можем сказать точно, за сколько времени таксист может доехать до пользователя. В этой задаче вам нужно разработать алгоритм, который будет определять, какие таксисты смогут **гарантированно** добраться до точки заказа за отведенное время, если примут его моментально. Таксист может стоять на месте или перемещаться по кругу **по часовой стрелке**, поэтому на момент вызова его позиция может измениться по сравнению с последней известной — он мог двигаться с любой скоростью, вплоть до максимально разрешенной. Точка заказа и точки, в которых находятся или могут находиться таксисты, — это точки на круге, перемещаться на вызов такси может только по часовой стрелке.

Формат ввода

В первой строке задано три целых числа: число событий N, длина круга L (в метрах), по которому перемещаются таксисты и на котором происходят заказы, максимальная скорость S с которой такси могут перемещаться по кругу.

В последующих N строках описываются события в следующем формате:

- `TAXI <timestamp> <taxi_id> <taxi_position>`. Команда, оповещающая о том, что таксист с номером $taxi\_id$ в момент времени $timestamp$ прислал координату своего местоположения. Местоположение определяется значением $taxi\_position$ — целым числом в диапазоне $[0, L)$, где 0 означает, что такси стоит в начале круга, а любое положительное число — дальность позиции таксиста от начала круга при движении по часовой стрелке.
- `ORDER <timestamp> <order_id> <order_position> <order_time>`. Команда, оповещающая о том, что в момент времени $timestamp$ пришёл заказ $order\_id$ на точку A, определяющуюся значением $order\_position$ — целым числом в диапазоне $[0, L)$ (где 0 означает, что заказ произошел в начале круга, а любое положительное число — дальность позиции вызова от начала круга при движении по часовой стрелке), до которого нужно **гарантированно** доехать за $order\_time$ секунд.

Поле $timestamp$ — это unix-like секундный таймстемп, все приходящие команды отсортированы по нему в порядке возрастания. Возрастание нестрогое: в нескольких последовательных командах могут быть одинаковые таймстемпы, но на момент события вызова мы рассматриваем только те события, которые нам известны (то есть не рассматриваем события, которые пришли после вызова, даже если у них такой же таймстемп).

Все заказы пронумерованы ($order\_id$), начиная с 0 в порядке их появления.

Ограничения:

- $N \leq 5000$;
- $0 < L \leq 100000$ (в метрах);
- $0 < S \leq 10$ (в метрах в секунду);
- $0 \leq order\_time \leq L$ (в секундах);
- $0 \leq taxi\_id < N$ (число от 0 до N).

Формат вывода

Программа должна вывести K строк, где K — количество команд $ORDER$.

В ответ на каждую команду $ORDER$ программа должна вывести строку с таксистами (набор $taxi\_id$, разделённых пробелами), которые гарантированно смогут добраться до точки A не более чем за $order\_time$. Если таких таксистов больше 5, то нужно вывести **5 любых**. Если ни один таксист не может гарантировано приехать в отведённое время, то в строке должен быть единственный идентификатор -1. Идентификаторы в одной строке не должны повторяться.

Пример

Ввод	Вывод
3 100 1 TAXI 1738300000 0 0 TAXI 1738300000 1 2 ORDER 1738300001 0 1 1	0

Примечания

Примечание к примеру.

Возможные позиции такси с id 0: $[0, 1]$. В таком случае на момент заказа таксист может добраться до него за 0 или 1 секунду, то есть он гарантированно может добраться за требуемую 1 секунду до точки вызова.

Возможные позиции такси с id 1: $[2, 3]$. То есть таксист может добраться до точки вызова не меньше чем за 98 секунд, поэтому он нам не подходит.