**Задание**

Поздравляем, вы добрались до первой задачи, в которой дизайн кода будет полностью вашего авторства. Это задание вы отправите на код-ревью и получите обратную связь от своих сокурсников. А ещё выступите в роли код-ревьюера для них.

Сохраните решение этой задачи, чтобы дальше загрузить его на платформу для ревью. Удачи!

Разработайте систему стимулирования чтения электронных книг. Для простоты будем считать, что книга всего одна, но её одновременно читают много людей. Нужно следить за прогрессом чтения у всех пользователей и выводить мотивирующие уведомления. А именно, программа должна обрабатывать следующие события:

* **READ user page** — сохранить факт того, что пользователь под номером **user** дочитал книгу до страницы **page**. Если ранее такой пользователь не встречался, нужно его добавить. Гарантируется, что в рамках одного пользователя номера страниц в соответствующих ему событиях возрастают.
* **CHEER user** — сообщить пользователю **user**, какая доля существующих пользователей, не считая его самого, прочитала меньшую часть книги, чем он. Если этот пользователь на данный момент единственный, доля считается равной 1. Если для данного пользователя пока не было ни одного события **READ**, доля считается равной 0, а сам пользователь не учитывается при вычислении долей для других пользователей до тех пор, пока для него не случится событие **READ**.

**Формат входных данных**

В первой строке вводится количество запросов Q — натуральное число, не превосходящее 10⁶. В следующих Q строках в соответствии с описанным выше форматом вводятся запросы. Гарантируется, что все вводимые числа целые и положительные, при этом номера пользователей не превосходят 10⁵, а номера страниц не превосходят 1000.

Пример ввода:

12

CHEER 5

READ 1 10

CHEER 1

READ 2 5

READ 3 7

CHEER 2

CHEER 3

READ 3 10

CHEER 3

READ 3 11

CHEER 3

CHEER 1

**Формат выходных данных**

Для каждого запроса **CHEER user** выведите единственное вещественное число от 0 до 1 — долю пользователей, прочитавших меньше страниц, чем **user**. Для вывода числа на экран используйте setprecision(6).

Пример вывода, соответствующий примеру ввода:

0

1

0

0.5

0.5

1

0.5

Пользователь 5 не учитывается при вычислении долей, потому что для него не произошло ни одного события **READ**. Пользователь 1 изначально был единственным, но в конце его обогнал третий, но не обогнал второй, поэтому он оказался продуктивнее 50% пользователей. Пользователь 3 изначально обгонял только второго и потому получал долю 50%, но в конце обогнал первого и получил долю 100%.

**Ограничения**

Четыре секунды на выполнение всех запросов. Все описанные в условии гарантии действительно справедливы для всех тестов, на которых будет запускаться ваша программа. Проверять корректность ввода не нужно. Гарантируется, что все вводимые числа целые и положительные. При этом номера пользователей не превосходят 10⁵, а номера страниц не превосходят 1000. Общее количество запросов не превосходит 10⁶.

**Что отправлять на проверку**

Подготовьте решение, соответствующее описанным требованиям и формату ввода-вывода. Сдайте его в тренажёре и сохраните себе в репозиторий. Это же решение нужно будет сдать на пир-ревью.

**Как будет тестироваться ваш код**

Мы отправим ну очень много запросов и проверим не только корректность, но и время их исполнения.

**Подсказка**

Храните два вектора. В первом по индексу n будет храниться номер страницы, до которой дочитал пользователь id, равным n. Во втором по каждому индексу m будет храниться количество пользователей, дочитавших как минимум до страницы m.