

Тестовое задание: Создание сервиса генерации числового мультипликатора

Описание задачи:

Необходимо разработать HTTP-сервис, который:

1. При инициализации принимает переменную rtp , где $rtp \in (0, 1.0]$.
2. По запросу клиента генерирует случайный числовой мультипликатор $multiplier \in [1.0, 10000.0]$.

API сервиса:

Сервис должен предоставлять HTTP-эндпоинт для генерации числовых мультипликаторов:

GET `http://localhost:64333/get`

Ответ возвращается в формате JSON и содержит сгенерированный числовой мультипликатор.

Пример ответа:

```
{  
  "result": 1.0  
}
```

Описание клиента:

Клиент работает с последовательностью чисел `sequence`, где каждое число $x \in [1.0, 10000.0]$.

Логика клиента следующая:

1. Для каждого числа x в последовательности `sequence` клиент запрашивает у сервиса числовой мультипликатор `multiplier`.
2. Клиент сравнивает `sequence` с x и применяет следующие правила:
 - Если $multiplier \leq x$, то x заменяется на 0.
 - Если $multiplier > x$, то x остается неизменным.
3. После обработки всей последовательности клиент вычисляет RTP по формуле:

$$RTP = \text{sum1} / \text{sum0},$$

где:

- `sum0` — количество всех начальных чисел последовательности.
- `sum1` — сумма чисел после преобразования.

P.S. $sum0$ на самом деле сумма значений клиента, которые необходимо умножить на $multiplier$, для простоты вычислений приняты равными 1, а $sum1$ - сумма этих значений, умноженных на $multiplier$ (в соответствии с п.2).

Пример:

Последовательность клиента:

`sequence = [2.0, 3.0, 100.0, 1.1, 1.1]`

Количество начальных чисел:

`sum0 = 5`

Сгенерированные мультипликаторы:

`multipliers = [1.8, 965.0, 1.0, 5.6, 1.2]`

Итоговая последовательность:

`transformed = [0, 3.0, 0, 1.1, 1.1]`

Сумма после преобразования:

`sum1 = 5.2`

Результат:

`RTP = 5.2 / 5 = 1.04`

Цель:

Сервис должен генерировать мультипликаторы таким образом, чтобы при обработке числовой последовательности на клиентской стороне значение RTP стремилось к значению `rtp` при количестве генераций мультипликатора $>10,000$.

Исходный код выложить в публичном репозитории github.

Сервис должен запускаться командой:

`go run . -rtp={значение}`