

Задача

Требуется реализовать pure-<ваш язык программирования> (использовать стандартные библиотеки из набора языка можно) http-сервер, цель которого выполнять сортировку ряда **целых** чисел и выдавать, в итоге, отсортированный ряд в форме http-ответа.

Сервер должен поддерживать два типа запросов:

- `/?concurrency=N&sort=<url>`, ответ на который - `jobid`, для созданной заявки сортировки ряда; `url` - http расположение ресурса с числами в форме **число, число, число,....**
- `/?get=<jobid>`, ответ на который - **json-объект** (`{ "state" = ? , "data" = ? }`), в котором содержится два поля:
 - `state=ready|queued|progress|eexist`
 - `ready` - набор отсортирован и находится в поле **data**;
 - `queued` - набор поставлен в очередь на сортировку, еще не готов, `data` - пустой массив [];
 - `progress` - набор находится в процессе сортировки, скачивание пока не допустимо, `data` - пустой массив [];
 - `eexists` - неверный `jobid`, `data` - пустой массив [];

Реализация

При старте http-сервера создается 2 потока, в первом запускается http-сервер (порт: 8888), второй используется как worker.

Поток с http сервером выполняет обработку http и отдает отсортированный контент. Скачивание исходного набора он не осуществляет.

Второй поток осуществляет действия по скачиванию и сортировке данных следующим образом:

1. Из очереди на сортировку выбирается новый `url`.
2. Данные скачиваются в локальный файл.
3. Данные из файла считываются (поток, не все сразу), превращаются в числа типа `int` в массив `int`;
4. Создается `N` потоков (из параметра `concurrency` для данного набора), в каждый передается `M/N` данных (`M` - общее количество данных в наборе, `N` - параметр запроса сортировки), где они сортируются `qsort`. Итоговые массивы сливаются в единый блок с помощью **алгоритма слияния**.
5. Данные помещаются на хранение в файл, память освобождается от данных.

При реализации необходимо предотвратить возможность гонки за ресурсы между потоками (локи, мьютексы).

PS: В задаче запрещается использовать: `nginx`, `apache`, `wget`, `curl (bin)`, etc. любые готовые программы, которые можно приклеить и облегчить жизнь, так же нельзя устанавливать сторонние библиотеки из репозитория типа `pear`, `ruri` и т.п.