# Тестовое задание

## Слияние логов

Имеется два файла с логами в формате **JSONL**, пример лога:

…

{"timestamp": "2021-02-26 08:59:20", "log\_level": "INFO", "message": "Hello"}

{"timestamp": "2021-02-26 09:01:14", "log\_level": "INFO", "message": "Crazy"}

{"timestamp": "2021-02-26 09:03:36", "log\_level": "INFO", "message": "World!"}

…

Сообщения в заданных файлах упорядочены по полю timestamp в порядке возрастания.

Требуется написать скрипт, который объединит эти два файла в один.

При этом сообщения в получившемся файле тоже должны быть упорядочены в порядке возрастания по полю timestamp.

К заданию прилагается вспомогательный скрипт на python3, который создает два файла "log\_a.jsonl" и "log\_b.jsonl".

Командлайн для запуска:

log\_generator.py <path/to/dir>

*\* Базовая задача*

*Ваше приложение должно поддерживать следующий командлайн:*

*<your\_script>.py <path/to/log1> <path/to/log2> -o <path/to/merged/log>*

*\* Задачка повышенной сложности (вместо работы с аргументами командной строки)*

*Необходимо реализовать графический интерфейс пользователя, который включает следующие элементы:*

1. *кнопка выбора первого \*.jsonl-файла*
2. *текстовое поле для отображения полного пути к первому выбранному \*.jsonl-файлу*
3. *кнопка выбора второго \*.jsonl-файла*
4. *текстовое поле для отображения полного пути к второму выбранному \*.jsonl-файлу*
5. *кнопка «Запуск», которая осуществляет запуск процесса обработки \*.jsonl-файлов*

*Графический интерфейс необходимо реализовать с использованием библиотеки PySide2 (либо PySide6).*

Готовое приложение необходимо выложить в репозиторий на GitHub (наличие коммитов во время разработки будет большим плюсом) и прислать ссылку на него (убедитесь, что репозиторий публичный).