

Эшелон Технологии

Группа разработки продуктов информационной безопасности

Тестовое задание соискателя на позицию
«Разработчик Go»



Правила

- Используйте для выполнения задачи язык программирования Go последней стабильной версии на момент работы над заданием.
- Используйте только стандартные библиотеки Go. Можно использовать внешние заимствования для логирования, реализации клиент-сервера (HTTP, gRPC), написания тестов. Нельзя использовать внешние заимствования, реализующие логику задания, например, алгоритм.
- Пожалуйста, выполняйте задание самостоятельно и не публикуйте решение в открытый доступ. Вы можете оформить решение как приватный репозиторий в GitHub и добавить права доступа к нему пользователя Github с именем @echelon-reviewer.

Критерии

Ваше задание будут оценивать разработчики нескольких команд по набору критериев:

- Соответствует ли программа требованиям задачи.
- Являются ли результаты выполнения программы корректными.
- Читаемость и логичность организации функций и пакетов в исходном коде.
- Старался ли разработчик следовать основным правилам написания кода Go, правила доступны на сайте официальной документации Go и наборе лучших практик, таких как «Uber Go Style Guide». Мы не будем придираться к деталям, но откажем соискателям, которые не придерживались никаких принципов организации кода, либо использовали организацию пакетов наследованную, например, от Java либо C++ проекта.
- Являются ли результаты выполнения программы корректными.
- Имеются ли тесты.
- Имеется ли написанный пайплайн для тестирования кода (GitHub Actions, Gitlab CI/CD);
- Получилось ли запустить программу с первого раза по инструкциям в README.md и получилось ли воспроизвести результаты.
- Нет ли явно избыточного использования памяти либо процессорного времени во время исполнения программы.
- Репозиторий либо папка с решением не содержат бинарных артефактов сборки.

Подача результата

Мы ожидаем от вас исходный код решённого задания с комментариями и инструкциями выполнения в файле README.md. Скомпилированные бинарные файлы отправлять не нужно. Варианты отправки:

- Опубликовать решение на github.com или gitlab.com.
- упаковать исходный код решения в архив `.tar.gz`.

Адрес репозитория либо файл с архивом отправляйте на n.frolova@npo-echelon.ru. Решённые задания принимаются только один раз.

Задание

Описание задания

Написать gRPC прокси-сервис для загрузки thumbnail'ов (превью видеоролика) с видеороликов YouTube. При повторном запросе на тот же видеоролик, сервис должен отдать закэшированный ответ (можно использовать рантайм кэш, но будет плюсом если будет использоваться временное хранилище, например sqlite). Так же предлагается написать клиентскую часть как утилиту командной строки, которой передается в качестве параметров ссылки на видеоролики. В командной утилите предусмотреть ключ `--async`, который позволяет скачивать большое количество файлов асинхронно.