

Домашняя работа 2. Реализация CLI, часть 1

24.01.2022

Дедлайн: 10:00 07.02.2022

Баллов: 10

Реализовать в тех же командах, в которых вы её проектировали, первую часть архитектуры Command-Line Interface из домашней работы 1, связанную с поддержкой Read-Execute-Print Loop и команд. Подстановки и пайпы пока не надо.

Должны поддерживаться:

- `cat [FILE]` — вывести на экран содержимое файла;
- `echo` — вывести на экран свой аргумент (или аргументы);
- `wc [FILE]` — вывести количество строк, слов и байт в файле;
- `pwd` — распечатать текущую директорию;
- `exit` — выйти из интерпретатора;
- если введено что-то, чего интерпретатор не знает — вызов внешней программы.

При этом:

- одинарные и двойные кавычки должны поддерживаться, хоть они ничем не отличаются без постановок: строка в кавычках — один аргумент;
- переменные окружения должны поддерживаться и передаваться внешнему процессу при запуске;
- для команд должны поддерживаться потоки вывода, ошибок и код возврата (для встроенных команд — на ваше усмотрение, для внешних процессов — что они вернут).

С технической точки зрения надо:

- чтобы оно собиралось и запускалось из консоли, а не только из вашей любимой IDE;
- очень желательно, чтобы оно работало и под Windows, и под Linux (не то чтобы это надо специально проверять, но старайтесь не использовать привязанных к ОС вещей);
- процесс сборки и запуска должен описан в README.md;
- должны быть юнит-тесты;
- должен быть настроенный CI, где эти юнит-тесты бы запускались (кстати, иметь CI и под Windows, и под Linux, и может даже под macOS было бы хорошей идеей);
- должны быть комментарии к каждому типу и каждому public-методу.

Сдавать в виде пуллреквеста с приличным названием в свой репозиторий (чтобы было понятно, какая это задача). Как сделаете, пишите в чат курса, что задача готова к проверке (не прикладывая ссылку на пуллреквест — чтобы у коллег из других команд было меньше соблазна посмотреть, как это всё делать).