

Задание 1. Пишем свою оболочку `microsha`.

Хорошая ли программа `sh`? А `bash`? Конечно, хорошие. Но большие. Ваша задача — написать программу, реализующую маааааленькое подмножество стандартных оболочек. Что она должна делать после своего запуска:

- Выдать приглашение к вводу, оно должно состоять из имени текущей директории и символа `>`, если она исполняется от непривилегированного пользователя или `!`, если от привилегированного.
- Считать стандартный ввод. Допускается любая длина входной строки, так что будьте аккуратны.
- Стандартный ввод разбивается на слова, разделённые пробелами или знаками табуляции.
- Переменными считаем все элементы окружения (`environments`, другое название — переменные среды). Они имеют строковый тип и могут содержать пробелы, знаки табуляции и другие символы, включая метасимволы.
- Имеется шесть метасимволов, имеющие тот же смысл, что и в стандартных оболочках, `>`, `<`, `|`, `*`, `?` и `"` (двойная кавычка).
- Символ `|` разбивает командную строку на компоненты конвейера. Как и в обычной оболочке каждая компонента представляет собой процесс, стандартный вывод которого соединяется со стандартным вводом следующего.
- Символ `$` в любом месте перед именем переменной требует замену этого имени со знаком `$` на строковое значение этого имени. При этом может вместо одного аргумента оказаться произвольное их число.
- Если в строке аргументов оказывается текст в двойных кавычках, то метасимволы `*`, `?`, `|`, `>`, `<` считаются обычными символами, пробелы и табуляции не считаются разделителями и весь текст от открывающей до закрывающей кавычки передаётся как есть в виде одного аргумента в нужный процесс.
- Первые слова каждого компонента конвейера (даже один) — имена команд. Команды могут быть внешними и внутренними. Внутренними командами являются:
 - `cd [arg]` — перейти в директорию с именем `arg`, при отсутствии аргумента — перейти в домашнюю директорию пользователя.
 - `pwd` — вывести на стандартный вывод полное имя текущей директории.
 - `time [cmd args]` — выполнить программу под именем `cmd` с аргументами `args`, после чего вывести на стандартный вывод ошибок три времени: время по обычным часам (`walltime`), время, которое запущенная программа провела в режиме системы и время, которое запущенная программа провела в режиме пользователя.
 - `echo [arg] ...` — вывести на стандартный вывод аргументы. Не думайте, что всё так легко, среди аргументов могут быть и переменные.
 - `set VAR=value` — присвоить переменной среды `VAR` значение строки `value`. Можно присваивать переменным среды и строки, содержащие пробелы и табуляции. Для этого их надо поместить в двойные кавычки.

Остальные команды — внешние и должны пытаться запустить указанную программу с указанными аргументами. Если запуск происходит в компоненте конвейера, то метасимволы `>` и `<` недопустимы.

- Метасимволы `*` и `?` в компоненте должны быть обработаны: каждая лексема, содержащая эти символы должна быть заменена на множество имён файлов, сопоставленных с ней. Не должна использоваться библиотека `glob` и ей подобные.
- После запуска командной строки `microsha` должна дожидаться завершения всех компонент и только после этого выдать приглашение на ввод очередной команды.

Проект не должен содержать ссылок на другие библиотеки. Очень рекомендую писать его сразу на C++.

Последний срок сдачи задачи №1 — 20 ноября.