

1. Считываем строку из input
  2. Заменяем переменные из данной строчки на их значения. Переменные со своими значениями находятся в пространстве имён. Всю работу осуществляет парсер. Парсер работает слева направо. Подстановка осуществляется жадно, то есть по максимальной длине. Подстановка в кавычках не осуществляется.
  3. После этого строка делится на секции по пайпам
  4. Назначаем абстракцию буфер, которая будет выполнять роль потока. Инициализируем буфер как входной поток.
  5. Проходимся по всем секциям
  6. Для очередной секции определяем ее тип. Тип секции может быть присваиванием переменной нового значения или команда с набором аргументов.
  7. Выполняем секции в соответствии с ее типом.
    - a. Если секция имеет тип присваивания, то обновляем множество переменных.
    - b. Если секция имеет тип команды с аргументами, то ищем выполняемую команду в пространстве команд и выполняем ее с указанными аргументами. Тут буфер передается как входной поток
  8. То, что должно содержаться в выходном потоке после выполнения секции, сохраняем в буфере. Выполняется переход к следующей секции
  9. Выводим содержимое буфера в выходной поток. Переводим программу в готовность читать следующую строку из входного потока
- A. Множество команд, где будет искаться выполняемая команда
- B. Множество переменных

#### Комментарии

- Синтаксис присваивания. Пробельных символов быть не должно, при их наличии то, что до первого пробела, будет восприниматься как названия команды. Слева от знака присваивания находится имя переменной, справа - строковое значение, которое может содержать подстановки.
- Синтаксис прочих команд. Имя команды, аргументы и пайпа разделяются пробелами. Лишние пробелы игнорируются.
- При некорректных командах или аргументах, программа выдаст соответствующее сообщение об ошибке и ждет следующего ввода.
- Подстановка выполняется один раз, рекурсивная подстановка невозможна.