

Вариант 3

Программе на вход подается текст (текст представляет собой предложения, разделенные точкой. Предложения - набор слов, разделенные пробелом или запятой, слова - набор латинских или кириллических букв, цифр и других символов кроме точки, пробела или запятой) Длина текста и каждого предложения заранее не известна.

Для хранения предложения и для хранения текста требуется реализовать структуры Sentence и Text

Программа должна сохранить (считать) текст в виде динамического массива предложений и оперировать далее только с ним. Функции обработки также должны принимать на вход либо текст (Text), либо предложение (Sentence).

Программа должна найти и удалить все повторно встречающиеся предложения (сравнивать их следует посимвольно, но без учета регистра).

Далее, программа должна запрашивать у пользователя одно из следующих доступных действий (программа должна печатать для этого подсказку. Также следует предусмотреть возможность выхода из программы):

- 1) Для каждого предложения вывести строку образец удовлетворяющую каждому слову в предложении. Строка условия содержит: символы, ? - 1 или больше любых символов, в начале и конце образца могут быть символы * - обозначающие 0 или больше символов. Например, для слов "Аристотель" и "Артишок", строка образец будет иметь вид "Ар???о?*".
- 2) Удалить все предложения, в которых нет заглавных букв в начале слова.
- 3) Отсортировать слова в предложении по количеству гласных букв в слове.
- 4) Для каждого предложения вывести количество одинаковых слов в строке.

Все сортировки и операции со строками должны осуществляться с использованием функций стандартной библиотеки. Использование собственных функций, при наличии аналога среди функций стандартной библиотеки, запрещается.

Каждую подзадачу следует вынести в отдельную функцию, функции сгруппировать в несколько файлов (например, функции обработки текста в один, функции ввода/вывода в другой). Также, должен быть написан Makefile.