**Обработка строк**

Создать программу обработки текста учебника по программированию с использованием классов: *Символ*, *Слово*, *Предложение*, *Абзац*, *Лексема*, *Листинг*, *Знак препинания* и др. Во всех задачах с формированием текста заменять табуляции и последовательности пробелов одним пробелом.Предварительно текст следует разобрать на составные части, выполнить одно из перечисленных ниже заданий и вывести полученный результат.

Текст должен содержаться в файле из, которого он считывается. Использовать для обработки классы StringBuilder и StringBuffer.

1. Найти наибольшее количество предложений текста, в которых есть одинаковые слова.

2. Вывести все предложения заданного текста в порядке возрастания количества слов в каждом из них.

3. Найти такое слово в первом предложении, которого нет ни в одном из остальных предложений.

4. Во всех вопросительных предложениях текста найти и напечатать без повторений слова заданной длины.

5. В каждом предложении текста поменять местами первое слово с последним, не изменяя длины предложения.

6. Напечатать слова текста в алфавитном порядке по первой букве. Слова, начинающиеся с новой буквы, печатать с красной строки.

7. Рассортировать слова текста по возрастанию доли гласных букв (отношение количества гласных к общему количеству букв в слове).

8. Слова текста, начинающиеся с гласных букв, рассортировать в алфавитном

порядке по первой согласной букве слова.

9. Все слова текста рассортировать по возрастанию количества заданной буквы в слове. Слова с одинаковым количеством расположить в алфавитном порядке.

10. Существует текст и список слов. Для каждого слова из заданного списка найти, сколько раз оно встречается в каждом предложении, и рассортировать слова по убыванию общего количества вхождений.

11. В каждом предложении текста исключить подстроку максимальной длины, начинающуюся и заканчивающуюся заданными символами.

12. Из текста удалить все слова заданной длины, начинающиеся на согласную

букву.

13. Отсортировать слова в тексте по убыванию количества вхождений заданного символа, а в случае равенства — по алфавиту.

14. В заданном тексте найти подстроку максимальной длины, являющуюся палиндромом, т. е. читающуюся слева направо и справа налево одинаково.

15. Преобразовать каждое слово в тексте, удалив из него все следующие (предыдущие) вхождения первой (последней) буквы этого слова.

16. В некотором предложении текста слова заданной длины заменить указанной подстрокой, длина которой может не совпадать с длиной слова.

17. Исключить из текста подстроку максимальной длины, начинающуюся и заканчивающуюся одним и тем же символом.18. Осуществить сжатие английского текста, заменив каждую группу из двух

или более рядом стоящих символов на один символ, за которым следует

количество его вхождений в группу. К примеру, строка hellowoooorld должна сжиматься в hel2owo4rld.