[Описание задачи 2](#_Toc401145828)

[Требования. 2](#_Toc401145829)

[Дополнительное задание 2](#_Toc401145830)

[Варианты заданий 3](#_Toc401145831)

# Описание задачи

Создать программу обработки текста учебника по программированию с использованием классов: Символ, Слово, Предложение, Знак препинания и др. Во всех задачах с формированием текста заменять табуляции и последовательности пробелов одним пробелом.

# Требования.

1. Текст учебника должен считываться с txt-файла.  
   Алгоритм следующий:
   1. Сделать бэкап оригинального файла копированием его в директорию для бэкапа (использование байтовых потоков ввода/вывода и средств класса **java.io.File**).  
      Структура проекта:  
       App/  
       |-src/  
       |-...  
       |-resource/  
       |-original/  
       |-book.txt  
       |-backup/  
       |-book.bak
   2. Считать текст с оригинального файла (использование символьных потоков ввода/вывода) и выполнить его парсинг.
      1. Парсинг необходимо осуществлять с помощью регулярных выражений и механизма работы с ними в Java.
      2. Регулярные выражения необходимо хранить отдельно от кода в properties-файле.
      3. Регулярные выражения получать с помощью ResourceBundle.
   3. Выполнить задание по вариантам с форматированным выводом в оригинальный файл, дописав его (использование символьных потоков ввода/вывода).
      1. Формат вывода смотреть в описании задания варианта.
2. Программа должна корректно реагировать на возникающие исключения (не существует файла, не закрыт поток и т.д.).

# Дополнительное задание

Уточнить у преподавателя о необходимости!

Реализовать возможность сохранения результатов работы парсера с помощью механизма сериализации.

# Варианты заданий

1. Вывести все предложения заданного текста в порядке возрастания количества слов в каждом из них. (Взять любой текст на 10 предложений)  
   Формат:  
   |====================================================================================|  
   | № | Sentence | Word's Count |  
   |=====+===============================================================+==============|  
   |1 | Word1? | 1|  
   |-----+---------------------------------------------------------------+--------------|  
   |2 | Word1 word2! | 2|  
   |-----+---------------------------------------------------------------+--------------|  
   |3 | Word1 word2 word3. | 3|  
   |-----+---------------------------------------------------------------+--------------|  
   |4 | Word1 word2 word3 word4? | 4|  
   |-----+---------------------------------------------------------------+--------------|  
   |5 | Word1 word2 word3 word4 word5. | 5|  
   |-----+---------------------------------------------------------------+--------------|  
   |6 | Word1 word2 word3 word4 word5. | 5|  
   |-----+---------------------------------------------------------------+--------------|  
   |7 | Word1 word2 word3 word4 word5. | 5|  
   |-----+---------------------------------------------------------------+--------------|  
   |8 | Word1 word2 word3 word4 word5. | 5|  
   |-----+---------------------------------------------------------------+--------------|  
   |9 | Word1 word2 word3 word4 word5. | 5|  
   |-----+---------------------------------------------------------------+--------------|  
   |10 | Word1 word2 word3 word4 word5 word6 word7 word8 word9 word10! | 10|  
   |====================================================================================|