Санкт-Петербургский государственный политехнический

Университет

Физико-Механический институт

Высшая школа прикладной математики и вычислительной физики

Алгоритм Хаффмана

Выполнил студент

гр. 5030102/20401

Данилова А.А.

Петербург

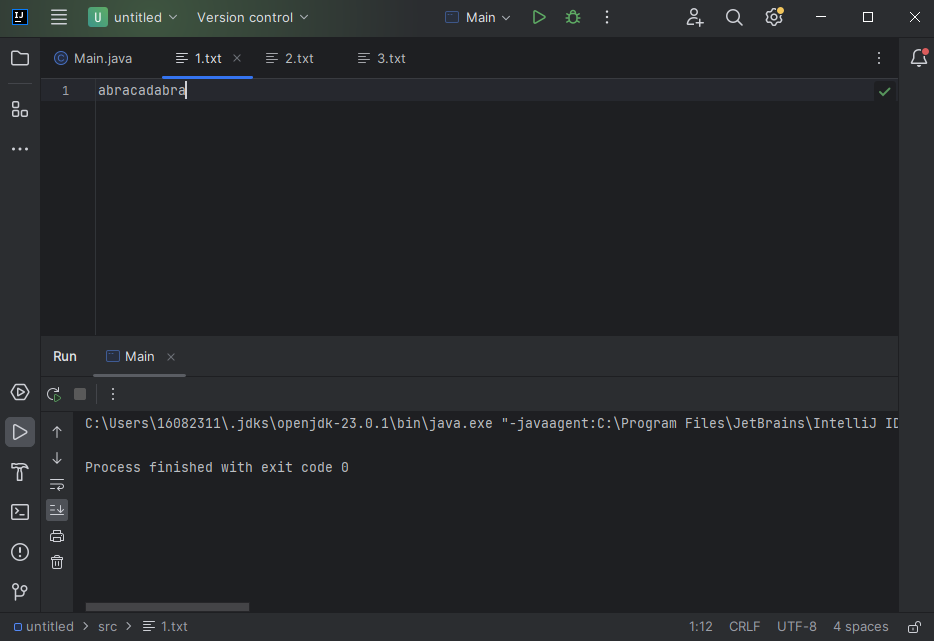
2024

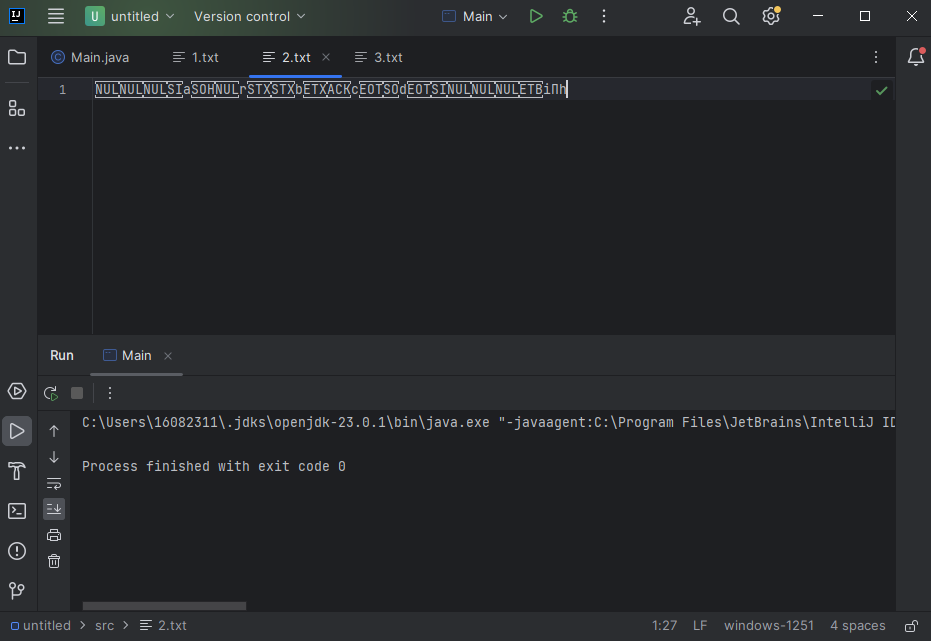
**Задача**

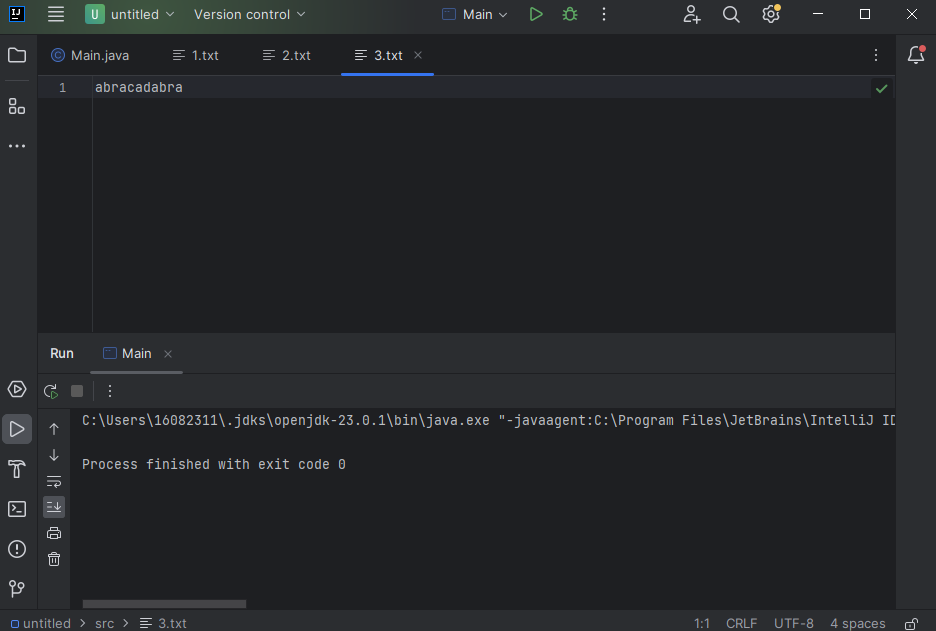
Реализовать алгоритм Хаффмана.

**Демонстрация работы алгоритма на примерах (1 фотография файл для кодирования, 2 фотография – закодированный файл, 3 фотография – раскодированный файл со 2 фотографии)**

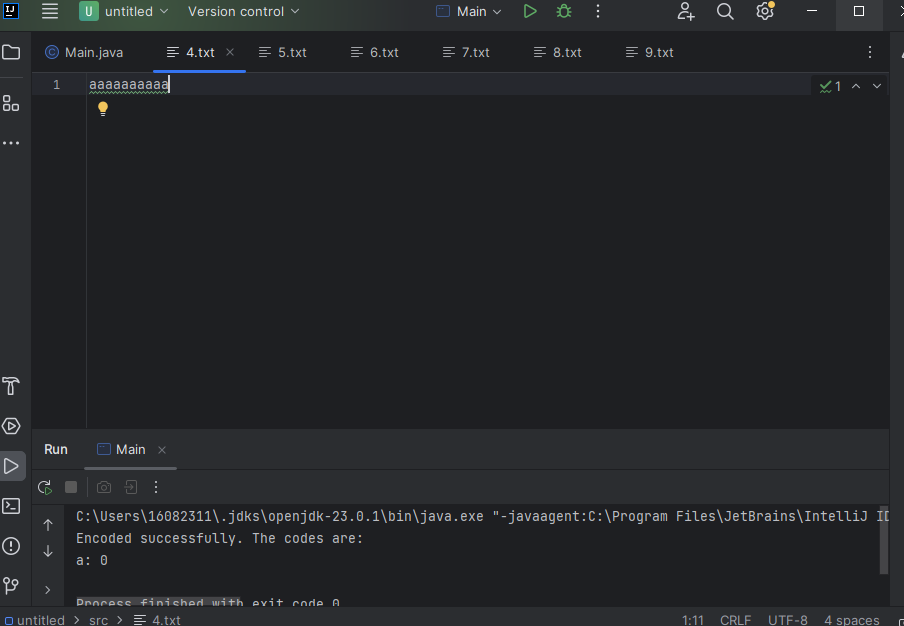
Пример 1

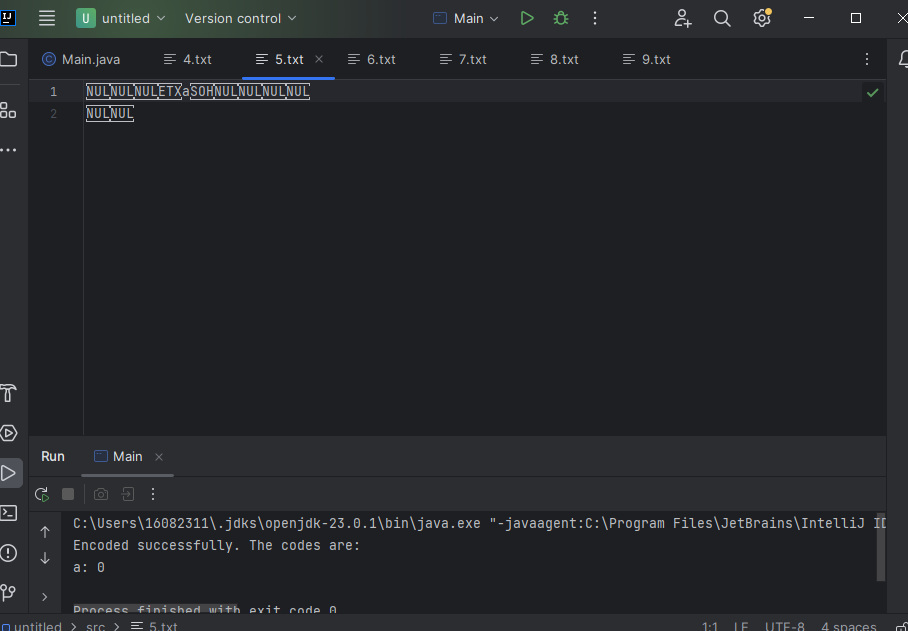


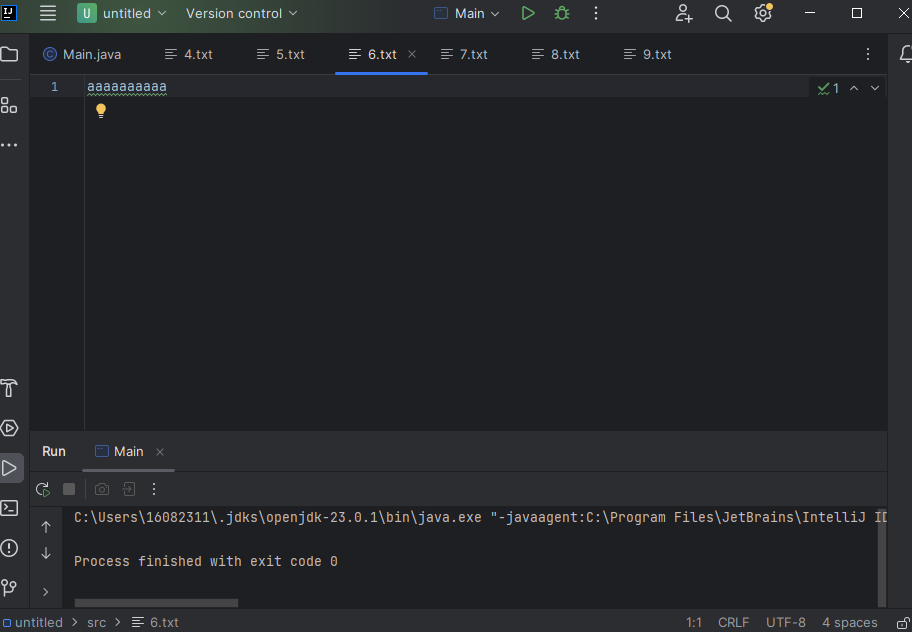




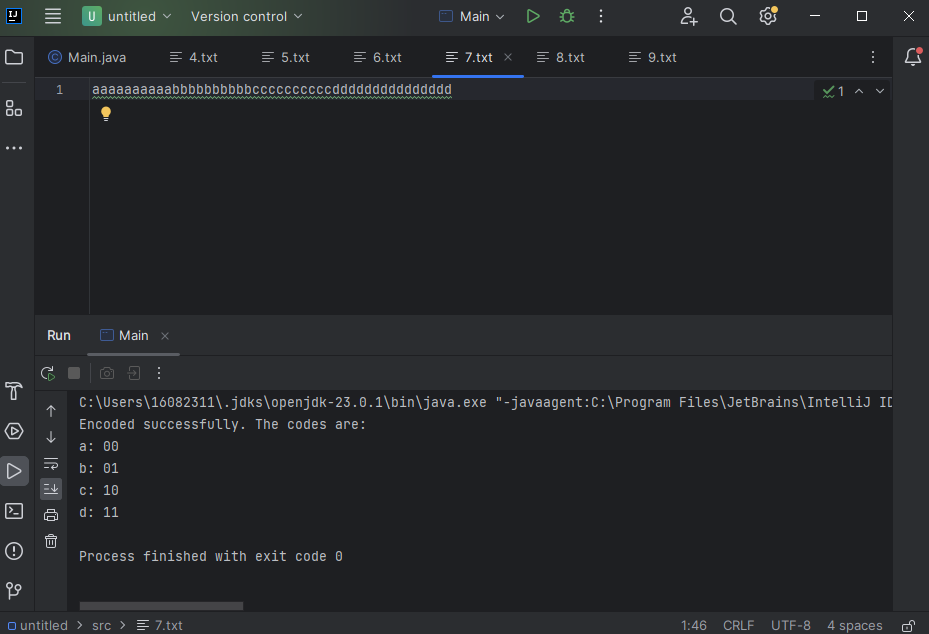
2 Пример

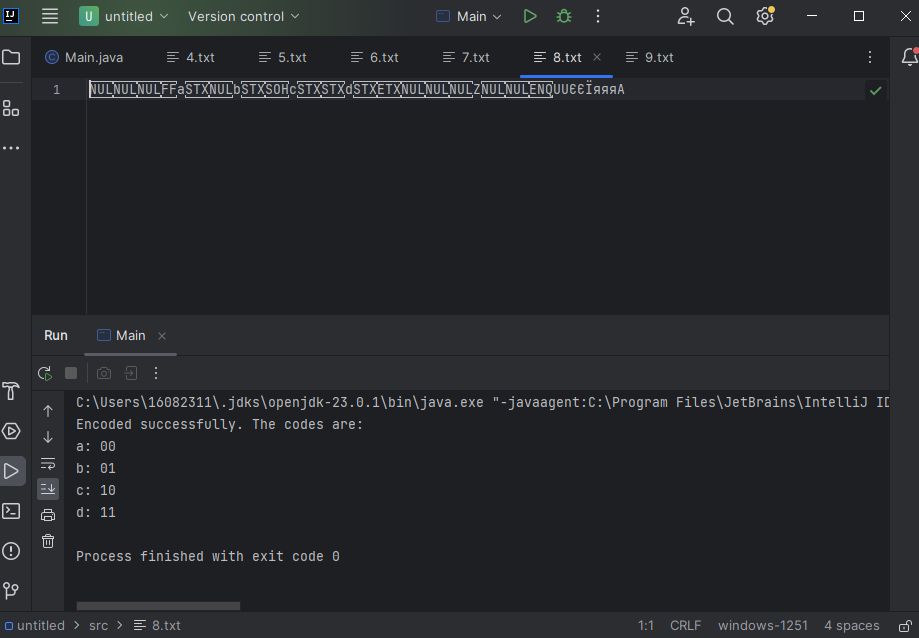


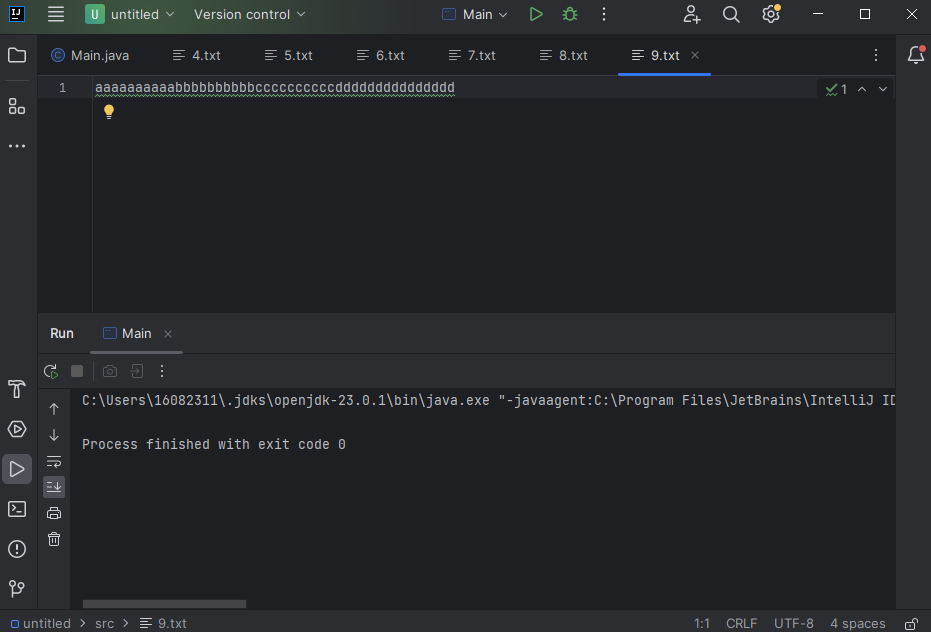




3 Пример







**Формат входных и выходных данных**

После кодирования получаем бинарный файл. Файл состоит из 2 больших частей:

\* Информация о том, каким байтам соответсвуют какие коды (кодировка осуществляется побайтово)

\* Закодированный текст

Информация о том, каким байтам соответсвуют какие коды:

В этой секции первые 2 байта это длина данной секции в байтах (не включая эти 2 байта, выделенные на длину). Затем идет сама информация о том, каким байтам соответствуют какие коды в виде <байт><длина кода в битах><биты кода>. Байт занимает, очевидно, 1 байт. Длина кода тоже занимает 1 байт, потому что длина кода не может превышать число кодируемых символов (глубина дерева не может быть больше числа его элементов - 1, то есть больше 255). Если закодированный байт занимает не целое количество байтов, то вконце дописываются нулевые биты. При этом при декодировании они будут проигнорированы. Это возможно потому что мы знаем точную длину кода в битах.

Закодированный текст:

Первые 8 байт этой секции описывают количество бит закодированного текста (не включая эти 8 байт, выделенные на длину). Далее идет сам закодированный текст. Если закодированный текст занимает не целое количество байт, то вконце дописываются нулевые биты. При этом при декодировании они будут проигнорированы. Это возможно потому что мы знаем точную длину закодированного текста в битах.

Примечание:

\* Почему на длину секции с информацией о кодах байтов выделено всего 2 байта:

Поскольку кодировка осуществляется побайтово, у нас не более чем 256 ззакодированных элементов. Для кодирования информации об одном байте нужно не более чем 10 байт (1 байт для кодируемого байта, 1 байт для длины код и не более 256 бит на сам код), так что длина секции не превышает 2560 и соответсвенно 2 байт достаточно.

\* Почему на длину секции с закодированным текстом выделено 8 байт:

В 8 байт влезает число вполть до 10 ^ 18. 10 ^ 18 бит это это примерно 113 тысяч терабайт. Что по сути означает, что ограничений на размер кодируемого файла нетВыходные данные – в обоих случаях выводится текст, в первом закодированный в байтах, в другом раскодированный в ascii символах.