**Алгоритм Хаффмана** — жадный алгоритм оптимального префиксного кодирования алфавита с минимальной избыточностью. Был разработан в 1952 году аспирантом Массачусетского технологического института Дэвидом Хаффманом при написании им курсовой работы. В настоящее время используется во многих программах сжатия данных.

Этот метод кодирования состоит из двух основных этапов:

* Построение оптимального кодового дерева.
* Построение отображения код-символ на основе построенного дерева.

На этапе построения оптимального кодового дерева, алгоритм строит двоичное дерево, в листьях которого лежат символы, используемые в кодируемом файле. Чем реже встречается символ, тем больше глубина его листа.

На втором этапе алгоритм присваивает символам двоичные коды в зависимости от их положения относительно корня дерева.

Таким образом, кодировка текста меняется из равномерной на неравномерную, в отдельных случаях помогая значительно сжимать файл.

**В программе присутствуют следующие классы:**

* **FileInputHelper** – класс отвечает за корректное считывание исходных файлов, преобразование бинарного кода в удобный для дальнейшей обработки вид.
* **FileOutputHelper** – класс отвечает за корректный вывод данных в файл. Имеет различные функции для вывода «обычного» и закодированного текста (по байту).
* **Node** – вспомогательный класс, описывающий одну из вершин бинарного дерева. Каждая вершина может хранить в семе символ, которому соответствует, количество его повторений (вес), ссылки на левого и правого «ребенка» либо nullptr вместо них.
* **PriorityQueue** – вспомогательный класс, реализующий приоритетную очередь для элементов класса Node.
* **HuffmanTree** – основной класс, объединяющий все предыдущие. Выполняет основной алгоритм – кодирует и декодирует файл.