**Федеральное государственное автономное образовательное учреждение**

**высшего образования**

**«Российский университет дружбы народов имени Патриса Лумумбы»**

Инженерная академия

*Кафедра механики и процессов управления*

**ОТЧЕТ**

|  |  |
| --- | --- |
| **По** | **Информатике и программированию** |

|  |  |
| --- | --- |
| **Направление:** | **Прикладная математика и информатика** |
|  | (код направления / название направления) |
| **Профиль:** | **Математические методы механики космического полета и анализ геоинформационных данных** |
|  | (название профиля) |

|  |  |
| --- | --- |
| **Тема:** | **Алгоритмы сжатия данных: реализация кодирования Хаффмана.** |
|  | (название лабораторной / курсовой) |

|  |  |  |  |
| --- | --- | --- | --- |
| **Выполнено:** | **Юнгова Анастасия Сергеевна** | | |
|  | (ФИО) | | |
| **Группа:** |  | **ИПМбд-02-23** |  |
| **№ студенческого:** | | 1132233510 |  |

**Москва, 2025**

Теоретическая сводка: алгоритм Хаффмана

**Кодирование Хаффмана** — это алгоритм сжатия данных без потерь, основанный на частоте появления символов.   
Он был предложен Дэвидом Хаффманом в 1952 году. Основная идея заключается в использовании переменной длины кодов для представления символов:   
чем чаще символ встречается, тем короче его код.  
  
**Основные шаги алгоритма:**

1. Подсчитать частоту каждого символа во входном сообщении.
2. Построить приоритетную очередь (очередь с минимальным приоритетом), где каждый узел — это символ и его частота.
3. Извлекать два узла с наименьшими частотами и объединять их в новый узел, сумма частот которых равна частоте нового узла.
4. Повторять шаг 3, пока в очереди не останется один узел — это и будет корень дерева Хаффмана.
5. Обход дерева позволяет задать бинарный код для каждого символа.  
     
   Изображение выглядит как домкрат

   Автоматически созданное описание

**Преимущества:**

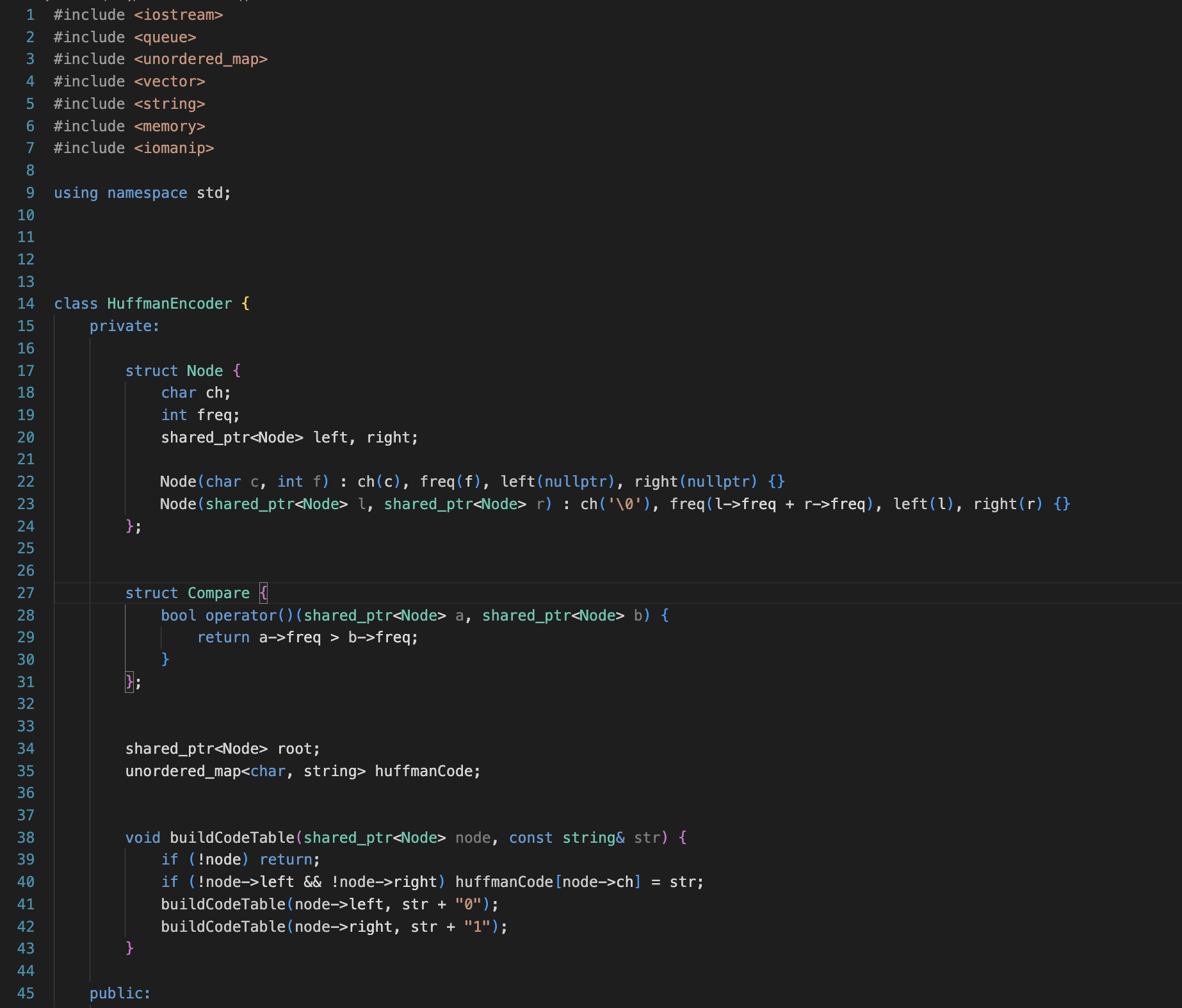
* Эффективное сжатие для текстов с неравномерным распределением символов.
* Простота реализации.

**Недостатки:**

* Неэффективно при равномерном распределении символов.
* Требует дополнительной информации (например, дерево или таблицу кодов) для декодирования.

**Алгоритм включает два этапа:**

* Кодирование: преобразование текста в битовую последовательность.
* Декодирование: восстановление исходного текста из битовой последовательности.

**Приложение  
  
**

**Изображение выглядит как текст, снимок экрана

Автоматически созданное описаниеИзображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание**

**Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание**[**Ссылка на гитхаб**](https://github.com/Seyell/kursovaya_4sem.git) **Список литературы:**1. Алгоритм Хаффмана // Википедия URL: <https://ru.wikipedia.org/wiki/Код_Хаффмана>

2. Теория с++ // Метанит URL: https://metanit.com/cpp/tutorial/5.1.php