**Федеральное государственное автономное образовательное учреждение высшего образования**

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

*Кафедра механики и процессов управления*

# Курсовая работа

По информатике и программированию

**Направление:** Прикладная математика и информатика

**Профиль:** Математические методы механики космического полёта и анализ геоинформационных данных

# Тема: Реализация сжатия Хаффмана

Выполнено студентом: Скруха Е. В + Киблер Р. М.

Группа: ИПМбд-02-23 + ИПМбд-01-23

№ студенческого: 1132232864 + 1132233518

# Москва, 2025

Алгоритм Хаффмана — это жадный алгоритм для **сжатия данных без потерь**, который присваивает символам строки **переменные по длине двоичные коды**, основываясь на их частотах. Более часто встречающиеся символы получают **короткие коды**, а редкие — **длинные**.

Как работает программа Шаги:

1. **Подсчёт частот** символов во входной строке.
2. **Создание дерева Хаффмана**:
   * Используется **минимальная приоритетная очередь**, чтобы всегда брать узлы с наименьшей частотой.
   * Узлы объединяются в пары, пока не останется один корневой.
3. **Построение кодов**:
   * Рекурсивно обходим дерево: налево — 0, направо — 1.
4. **Кодирование** строки по полученным кодам.
5. **Декодирование** бинарной строки обратно в текст.
6. **Очистка памяти** — удаление всех узлов дерева.

Пример: как работает дерево Хаффмана

Предположим, у нас есть текст: "abcde" Частоты:

* a: 1
* b: 1
* c: 1
* d: 1
* e: 1

Будет построено дерево:

(\*)

/ \

(\*) (\*) / \ / \ a b c (\*) / \ d e

Теперь, если идти от корня до символа:

* a → 000
* b → 001
* c → 01
* d → 10
* e → 11