**Лабораторная работа 6**

**Арифметическое кодирование**

**Цель работы**: изучить принципы арифметического кодирования, применяемого для сжатия данных, и разработать алгоритм кодирования и декодирования текста на его основе. В процессе работы студенты научатся оценивать эффективность сжатия с помощью арифметического кодирования и сравнивать его с другими методами.

**Теоретическая часть**

Арифметическое кодирование — это метод сжатия данных, который преобразует всё сообщение в одно число, лежащее в диапазоне от 0 до 1. Вместо кодирования каждого символа фиксированным количеством бит, как это делается в других методах, арифметическое кодирование обрабатывает все символы вместе, постепенно уточняя интервал для закодированного числа.

**Основные принципы:**

- Инициализация диапазона:

Начальный интервал для кодирования — это отрезок [0, 1). Весь процесс состоит в том, чтобы постепенно сужать этот интервал по мере обработки каждого символа, основываясь на вероятности появления символов.

- Разбиение интервала:

Интервал делится пропорционально вероятностям символов в исходном сообщении. Чем чаще символ встречается, тем больший отрезок он занимает.

- Кодирование символов:

По мере обработки каждого символа, текущий интервал делится на более мелкие участки, и выбирается подинтервал, соответствующий символу.

- Финальная кодировка:

В итоге сообщение кодируется одним числом, которое принадлежит к финальному подинтервалу.

**Пример 1:**

Пусть у нас есть строка "ABB" и следующие вероятности символов:

A: 0.5

B: 0.5

Начальный интервал: [0, 1).

После обработки первого символа (A) интервал делится на два:

A: [0, 0.5)

B: [0.5, 1)

Следующий символ (B) сужает интервал для B:

* A: [0.5, 0.75)
* B: [0.75, 1)

Таким образом, строка "ABB" кодируется как число, попадающее в интервал [0.75, 0.875).

**Пример 2 (более подробный) для строки "AB AA B":**

1. **Определение вероятностей символов**

Мы имеем строку "AB AA B", которая содержит 7 символов: *A* встречается 3 раза. *B* встречается 2 раза. *Пробел* встречается 2 раза. Общая длина строки – 7 символов. Соответственно, вероятности символов: P(A) = 3/7 ≈ 0.4286, P(B) = 2/7 ≈ 0.2857, P(пробел) = 2/7 ≈ 0.2857.

Теперь делим интервал [0, 1) на пропорциональные части, исходя из вероятностей:

Интервал для A: [0.0, 0.4286)

Интервал для B: [0.4286, 0.7143)

Интервал для пробела: [0.7143, 1.0)

1. **Кодирование строки "AB AA B":**

Для кодирования строки будем сужать интервал, каждый раз обновляя его границы в зависимости от символов.

**Шаг 1: Кодируем символ "A"**

Начальный интервал составлял: [0.0, 1.0). Интервал для символа "A" = [0.0, 0.4286). Значит, наш новый интервал для строки = [0.0, 0.4286)

**Шаг 2: Кодируем символ "B"**

Текущий интервал, в соответствии с предыдущим шагом, составляет: [0.0, 0.4286). Делим этот интервал на части согласно вероятностям:

- Интервал для символа "A" = [0.0, 0.1837)

- Интервал для символа "B" = [0.1837, 0.3061)

- Интервал для пробела = [0.3061, 0.4286)

Так как текущий символ "B", то выбираем интервал [0.1837, 0.3061).

**Шаг 3: Кодируем пробел**

Текущий интервал составляет: [0.1837, 0.3061).

Делим этот интервал:

- Интервал для "A" = [0.1837, 0.2326)

- Интервал для "B" = [0.2326, 0.2704)

- Интервал для пробела = [0.2704, 0.3061)

Текущий символ «пробел» поэтому, выбираем интервал [0.2704, 0.3061).

**Шаг 4: Кодируем символ "A"**

Текущий интервал: [0.2704, 0.3061).

Делим этот интервал:

- Интервал для "A" = [0.2704, 0.2851)

- Интервал для "B" = [0.2851, 0.2956)

- Интервал для пробела = [0.2956, 0.3061)

Так как текущий Символ "A", то выбираем интервал [0.2704, 0.2851).

**Шаг 5: Кодируем символ "A"**

Текущий интервал: [0.2704, 0.2851)

Делим этот интервал:

- Интервал для "A" = [0.2704, 0.2765)

- Интервал для "B" = [0.2765, 0.2807)

- Интервал для пробела = [0.2807, 0.2851)

Символ — "A", выбираем интервал [0.2704, 0.2765).

**Шаг 6: Кодируем пробел**

Текущий интервал: [0.2704, 0.2765)

Делим этот интервал:

- Интервал для "A" = [0.2704, 0.2729)

- Интервал для "B" = [0.2729, 0.2749)

- Интервал для пробела = [0.2749, 0.2765)

Текущий символ — пробел, выбираем интервал [0.2749, 0.2765).

**Шаг 7: Кодируем символ "B"**

Текущий интервал: [0.2749, 0.2765)

Делим этот интервал:

- Интервал для "A" = [0.2749, 0.2755)

- Интервал для "B" = [0.2755, 0.2760)

- Интервал для пробела = [0.2760, 0.2765)

Текущий символ — "B", выбираем интервал [0.2755, 0.2760).

**Финальный интервал**

Закодированное значение будет ***любым числом*** в интервале [0.2755, 0.2760).

Например, можно выбрать число посередине: 0.27575

Строка "AB AA B" была закодирована в число **0.27575**. Это число является уникальным для данной строки и всех символов, входящих в неё. Для декодирования этого числа можно обратно использовать тот же процесс сужения интервалов.

**Преимущества:**

1. Арифметическое кодирование достигает практически теоретического предела сжатия.
2. Кодировка более эффективна, чем, например, код Хаффмана, особенно для длинных сообщений.

**Практическая часть**

**Задачи лабораторной работы:**

1. Реализовать алгоритм арифметического кодирования для произвольной строки.
2. Провести тестирование программы на различных исходных данных. Сформулировать выводы, для каких данных арифметическое кодирование более эффективно.
3. Провести тестирование программы на различных строках и наборе вероятностей. Сравнить длину сжатого сообщения с исходным.
4. Вычислить отношение сжатия:

Сompress=Voriginal / V compressed

1. Оценить эффективность арифметического кодирования для различных наборов данных (текст, случайные данные).
2. Оценить эффективность сжатия по сравнению с другими методами (например, RLE или кодирование Хаффмана). См. предыдущие лабораторные работы.

***p.s. Перед защитой работы вручную вычислить код для строки «ABCABEAAFB» и сравнить результат с программным вычислением.***

**Ход работы:**

1. **Разработка программы:**
   * **Входные данные:** Строка для кодирования и частоты (вероятности) появления символов.
   * **Алгоритм кодирования:**
     + Инициализация интервала [0, 1).
     + Для каждого символа:
       - Деление текущего интервала в соответствии с вероятностями.
       - Выбор подинтервала для символа.
     + Получение конечного числа, представляющего закодированное сообщение.