

Выполнил(а) Денисова Алёна, № группы P3131, оценка
Фамилия И.О. студента не заполнять

Название статьи/главы книги/видеолекции Сжатие данных LZW			
ФИО автора статьи (или e-mail) kmoseenk	Дата публикации "5" октября 2021 г.	Размер статьи 1080	
Прямая полная ссылка на источник или сокращённая ссылка (bit.ly, tr.im и т.п.) https://habr.com/ru/companies/otus/articles/581728/			
Теги, ключевые слова или словосочетания Сжатие данных, кодирование и декодирование информации			
Перечень фактов, упомянутых в статье (минимум три пункта) <ol style="list-style-type: none"> 1. LZW – метод сжатия данных, основывающийся на многократном повторении символов в файле. 2. Основные этапы алгоритма: 1) начать с исходной модели; 2) читать данные по частям; 3) обновлять модель и кодировать данные по мере продвижения. 3. Чтобы закодировать подстроку, в выходной файл нужно записать только одно число, соответствующее индексу этой подстроки в словаре. 4. Сначала LZW использует словарь из 256 символов в качестве стандартного набора символов. 5. Затем алгоритм считывает данные по 8 бит за раз и кодирует их в виде числа (индекс в словаре). 6. Встречая новую подстроку, LZW добавляет ее в словарь; встречая уже «знакомую» подстроку, он считывает новый символ и выполняет его конкатенацию с этой подстрокой, получая новую. 7. Обычно для словаря задается максимальное количество записей, чтобы процесс не исчерпал память. 8. Программы кодирования и декодирования должны начинаться с одного начального словаря. 9. Декодер считывает индекс и выводит подстроку с этим индексом. Первый символ подстроки конкатенируется с текущей строкой, эта конкатенация добавляется в словарь. Строка становится текущей рабочей строкой и процесс повторяется. 10. В случае, если вызывается символ, который еще обрабатывается, берется уже полученная подстрока и ее первый символ конкатенируется с самим собой. 			
Позитивные следствия и/или достоинства описанной в статье технологии (минимум три пункта) <ol style="list-style-type: none"> 1. Для декодирования не требуется словарь – идентичный созданному во время сжатия словарь восстанавливается в процессе декодирования. 2. Алгоритм не требует вычисления вероятностей встречаемости символов или кодов. 3. Алгоритм не вносит искажений в исходный файл. 			
Негативные следствия и/или недостатки описанной в статье технологии (минимум три пункта) <ol style="list-style-type: none"> 1. LZW не проводит анализ входных данных и поэтому не является самым оптимальным. 2. Ограниченность размера словаря. Его можно увеличить, но тогда произойдет и замедление процесса поиска совпадений. 3. Кодирование одиночных символов крайне неэффективно. 			
Ваши замечания, пожелания преподавателю или анекдот о программистах¹ Диалог двух студентов во время сдачи лабы: Студент 1: Ты любишь хомячков? Студент 2: Да, они похожи на втшников. Студент 1: Чем? Студент 2: Редко живут больше двух лет. P.S. А за что ставились баллы за посещаемость первой лекции? Я была, и форму заполнила, а баллы не стоят((

¹ Наличие этой графы не влияет на оценку