Контрольное домашнее задание

Постановка задачи

(1) Разработать на языке C++ программу, реализующую алгоритмы сжатия данных без потерь, получение архивного файла из исходного и разархивированного файла из архивного (упаковка файла и распаковка архива)

1. алгоритм Хаффмана (простой) - студенты, у которых логин заканчивается на четное число.
2. алгоритм Шеннона-Фано (простой), студенты, у которых логин заканчивается на нечетное число
3. алгоритм Лемпеля-Зива LZ77 (со скользящим окном),
4. алгоритм Лемпеля-Зива-Велча LZW ***Бонус-задача.** Включение данного алгоритма в исследование не обязательно. Его реализация и включение в состав экспериментального исследования позволяет получить дополнительные баллы (см. раздел Оценивание).

Материалы с описанием алгоритмов имеются в LMS. Списки источников также есть в ЛМС (семинары 6 и 7).
Первые этапы алгоритмов были реализованы вами в контестах 6 и 7.

(2) Провести вычислительный эксперимент для исследования эффективности реализованных алгоритмов сжатия без потерь (упаковка и распаковка) для файлов разного типа.
Для проведения эксперимента с алгоритмами сжатия без потерь необходимо использовать набор из **N** файлов различных типов с именами 1.*...N.*.
Вы должны самостоятельно подобрать исходные файлы примерно одного размера, желательно не менее 12000 Кб.
В наборе должны присутствовать файлы

- текстовый .txt
- документ Word .docx
- презентация .pptx
- документ .pdf
- исполняемый файл .exe или библиотечный .dll
- цветное изображение .jpg
- изображение черно-белое или в градациях серого .jpg
- цветное изображение .bmp
- изображение черно-белое или в градациях серого .bmp
- файлы других форматов.

Все исходные файлы скопируйте в папку DATA, которая должна находиться

там же, где исполняемый файл. Программа должна автоматически обработать все файлы из этой папки и в нее же записать все полученные результаты.

Форматы имен файлов:

1. исходный файл <name>.*
2. метод упаковки, использующий алгоритм Хаффмана, упакованный файл <name>.haff
3. метод распаковки, использующий алгоритм Хаффмана, распакованный файл <name>.unhaff
4. метод упаковки, использующий алгоритм Шеннона-Фано, архивированный файл <name>.shan
5. метод распаковки, использующий алгоритм Шеннона-Фано, разархивированный файл <name>.unshan
6. метод упаковки, использующий алгоритм LZ77 (
   - размер скользящего окна 5 Кб, размер словаря 4 Кб архивированный файл <name>.lz775,
   - размер скользящего окна 10 Кб, размер словаря 8 Кб архивированный файл <name>.lz7710,
   - размер скользящего окна 20 Кб, размер словаря 16 Кб архивированный файл <name>.lz7720,
7. метод распаковки, использующий алгоритм LZ77;
   - размер скользящего окна 5 Кб, размер словаря 4 Кб архивированный файл <name>.unlz775,
   - размер скользящего окна 10 Кб, размер словаря 8 Кб архивированный файл <name>.unlz7710,
   - размер скользящего окна 20 Кб, размер словаря 16 Кб архивированный файл <name>.unlz7720,
8. *****метод упаковки, использующий алгоритм LZW, архивированный файл <name>.lzw
9. *****метод распаковки, использующий алгоритм LZW, разархивированный файл <name>.unlzw.

Вычислить:

- *энтропию* исходных файлов; определяется общее количество различных символов w, вычисляется их частотная встречаемость w_i в файле, и энтропия файла по формуле

$$H = -\sum_{i=1}^{m} w_i log_2 w_i;$$

  значения близкие к 1 характеризуют данный файл, как файл с близкой к равночастотной встречаемостью символов;
- коэффициент сжатия как отношение размера сжатого файла к размеру исходного файла.

Измерить для каждого файла и для каждого алгоритма:

- время упаковки;
- время распаковки.

Время измерять в тактах ЦП или в наносекундах (как на учебной практике

летом 2019 года) (если значение в микросекундах равно нулю). Для получения достоверных результатов упаковку и распаковку каждого файла каждым методом выполнить не менее 10 раз, после чего вычислить среднее время работы каждого алгоритма на каждом файле. Количество экспериментальных измерений времени **не менее**

(10 раз * 4 * 2 (или 5 * 2)) * N файлов = **80N (или 100N)**

(учтены алгоритмы упаковки/распаковки LZ77 с разным размером окна).

(3) Подготовить отчет по итогам работы. В отчете необходимо ***явно*** указать, какие части задания были сделаны, а какие нет

(4) Результаты работы надо *загрузить в ЛМС* (проект КДЗ) в виде архива, содержащего:

1. Отчет по работе (в форматах pdf или doc/docx),
2. Таблицы и графики
3. Исходные коды проекта
4. Исполняемый файл
5. Исходные файлы и архивы (если получится их загрузить в ЛМС. Если не получится. принесите их на защиту)

Отчет по работе

Примерное содержание отчета о работе:

1. Титульный лист
2. Оглавление
3. Краткая постановка задачи с указанием выполненных и невыполненных пунктов.
4. Описание алгоритмов и использованных структур данных
5. Описание реализации алгоритмов (форматы сжатых файлов для всех алгоритмов, особенности упаковки / распаковки LZ77, методы работы с битами/ байтами при сжатии / распаковке и др.),
6. План эксперимента
7. Описание программной реализации эксперимента, использованных дополнительных инструментов (например, если использовались скрипты автоматизации)
8. Диаграмма классов
9. Использованные аппаратные средства для проведения эксперимента
10. Результаты экспериментов - таблицы и графики (подробнее см. далее в этом документе),
11. Сравнительный анализ алгоритмов по эффективности сжатия файлов разных типов, по скорости работы
12. Заключение (основные выводы)

13. Использованные источники

Таблицы

Результаты выполнения экспериментов *необходимо* оформить в виде таблиц.

Таблица 1 - частоты появления символов в файле

файл	1	2	...	N
символ	Частота появления символа			
0				
1				
2				
...				
255				
Энтропия H				

где
H - энтропия исходного файла,

В таблице 1 надо показать отношение (количество появлений символа)/ (общее количество символов) для каждого файла.

Таблица 2 - Коэффициенты сжатия файлов

Имя файла	S1	H	Алгоритм Хаффмана или Шеннона-Фано		Алгоритм LZ77, окно 5 Кб		Алгоритм LZ77, окно 10 Кб		Алгоритм LZ77, окно 20 Кб	
			S2	K	S2	K	S2	K	S2	K
1										
2										
...										

где
H - энтропия исходного файла,
K - коэффициент сжатия,
S1 - размер исходного файла,
S2 - размер сжатого файла.

Таблица 3 - Время упаковки / распаковки файлов

Имя файла	S1	H	Алгоритм Хаффмана или Шеннона-Фано		Алгоритм LZ77, окно 5 Кб		Алгоритм LZ77, окно 10 Кб		Алгоритм LZ77, окно 20 Кб	
			tu	tp	tu	tp	tu	tp	tu	tp
1										
2										
...										

где
tp - время упаковки (nanoseconds),
tu - время распаковки (nanoseconds).

Каждая строка таблиц 2 и 3 содержит результаты выполнения эксперимента для одного из файлов тестового набора (всего N).

Графики

Отчет *должен содержать* следующие графики и иллюстрации.

1. Диаграмма (одна общая для всех файлов или для каждого файла своя) распределения частот встречаемости символов (байтов) для файлов в следующем формате (по оси OX - значения байта, всего 256 значений; по оси OY - частота появления символа в интервале [0; 1], легенда - имя файла и значение его энтропии):

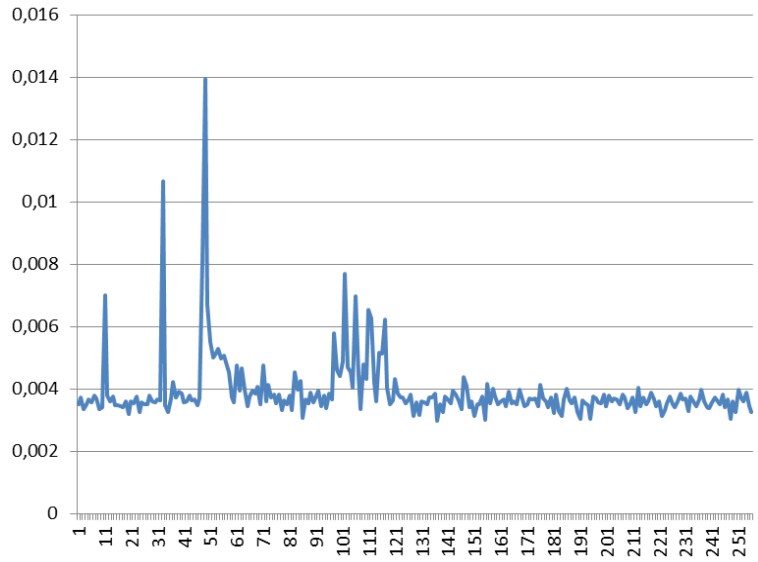

Рисунок 1. Пример диаграммы распределения относительных частот символов для одного файла

2. Столбчатые диаграммы, отражающие
   - коэффициент сжатия каждого файла для каждого алгоритма (Ось OX - номер (имя) файла, ось OY - коэффициент сжатия, легенда - название алгоритма),
   - время упаковки каждого файла для каждого алгоритма,
   - время распаковки каждого файла для каждого алгоритма.

Пример диаграмм, на которых приводятся данные коэффициента сжатия (цифры случайные, для примера построения диаграмм):

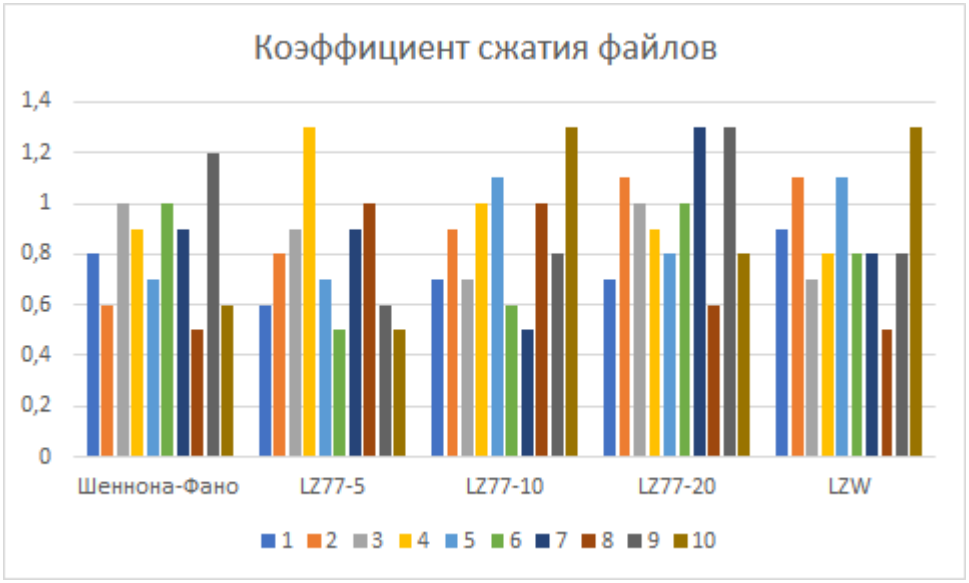

Рисунок 2. Пример диаграммы эффективности сжатия разных алгоритмов на разных файлах

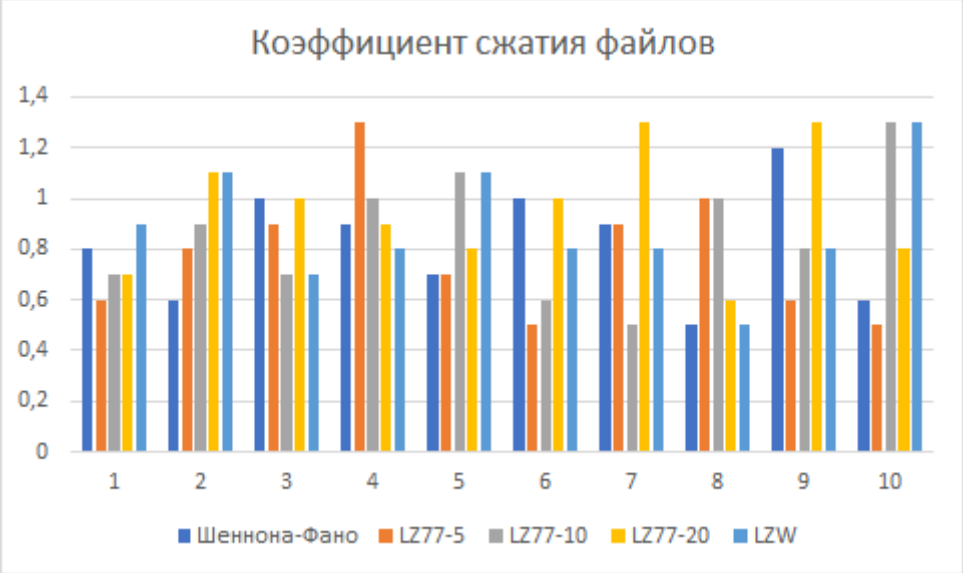

Рисунок 3. Пример диаграммы эффективности сжатия разных алгоритмов на разных файлах

Как видно из рисунков, диаграммы можно представить в разных вариантах. Построенные вами диаграммы должны позволять сравнить эффективность (коэффициент сжатия), время работы алгоритмов упаковки и время работы алгоритмов распаковки на предложенных файлах.

Позволяют ли такие диаграммы судить о содержании (формате) тестовых файлов?

> **Подсказка:** Стиль оформления в вашей работе может отличаться от используемого в настоящем задании. Мы представляем не все необходимые графики, а только примеры. **Важно**, чтобы *оси графиков* были соответствующим образом оцифрованы и подписаны, единицы измерения должны быть указаны, приведены легенды графиков, разъясняющие смысл раскраски и начертания линий и т.д.

Приветствуются эксперименты с разными видами и способами представления информации *в дополнение к требуемым*. Можно ли наглядно показать какие-то зависимости в трех измерениях? Можно ли задействовать размер и форму точек на графиках?

Обратите внимание, что для небольших файлов с большим количеством символов *архив может быть больше исходного файла* по размеру. Почему? Ваши эксперименты подтверждают или опровергают это наблюдение?

Отчет *может* содержать дополнительные таблицы/графики, которые студент сочтет информативными и полезными в рамках задачи.

Исходные коды

В файле main.cpp указать в комментариях (в самом начале файла):

```
// КДЗ по дисциплине Алгоритмы и структуры данных, 2019-2020 уч.год
// ФИО студента, группа БПИ-XXX, дата (XX.XX.2020)
// Среда разработки,
// Состав проекта (файлы *.cpp и *.h)
// Что сделано (сжатие и распаковка методом Хаффмана / сжатие и распаковка
методом Шеннона - Фано, сжатие и распаковка методом LZ77, сжатие и
распаковка методом LZW, проведен вычислительный эксперимент, построены
таблицы и графики, для измерения времени выполнения использовалось XXX,
оформлен отчет)
// Что не сделано (см. список выше)
```

В коде должно быть достаточно комментариев для того, чтобы его мог понять другой программист (и вы сами не забыли, что и как сделали).

Сроки выполнения и защиты КДЗ

Срок загрузки проекта и отчета в ЛМС — 03.04.2020 (пятница), 10:30:00, защита работ — во время семинаров 03 и 04 апреля 2020 г.
Защита проводится **только** при условии загрузки отчёта по КДЗ в ЛМС.
Защита проводится по предварительно составленному графику.

Для опоздавших срок загрузки отчета - до 10:30:00, 07.04.2020 (вторник).
При опоздании загрузки проекта в ЛМС **штраф 2 балла**.

Оценивание

Оценка за работу выставляется по итогам *очной защиты* проекта преподавателю / учебному ассистенту

Составляющая работы	**Балл (макс)**
Реализация упаковки/распаковки алгоритмом Хаффмана или Шеннона-Фано	2
Реализация упаковки/распаковки алгоритмом LZ77	2
Реализация эксперимента, измерение времени работы	2
Анализ результатов, полнота отчета (описание алгоритмов и структур данных, особенностей реализации, наличие всех графиков, осмысленность выводов)	2
Защита КДЗ (пояснение решений, ответы на вопросы)	2
*Реализация бонусного алгоритма LZW и выполнение всех элементов КДЗ для этого алгоритма (пункт не является обязательным)	+3 балла
Итого	**10 +*3**

Заключительные замечания

1. При выполнении домашних работ от вас требовалось только найти коды для символов кодируемого файла. При этом вы работали со строками, а выходной файл не кодировали.

При решении КДЗ надо реализовать реальные кодировщик и распаковщик. Т.е. такие, которые могут быть применимы для сжатия файлов.
Рекомендуем входной файл для упаковки рассматривать как поток байтов. В действительности, для обеспечения хорошего сжатия надо *работать с выходными файлами на битовом уровне*. По [ссылке] доступен пример работы с битами. От студента при выполнении КДЗ требуется самостоятельно разобраться в технической стороне вопроса.
Мы сознательно не ограничиваем вас в выборе того или иного способа работы с файлом, чтобы Вы:

- могли проявить творческую активность,
- проявили самостоятельность (и задумались над тем, каким образом будет лучше организовать такую работу),
- имели меньше возможностей для заимствования.

2. За один день (и даже за три дня) работу хорошо выполнить **невозможно**! Это надо иметь в виду.

3. Плагиат строго наказываем, как и всегда.

Желаем успеха!

Разработано преподавателями ДПИ ФКН Р. З. Ахметсафиной, А. А. Мицюком и М. В. Ульяновым

Версия 10.03.2020