**Практическая работа 8**

**Архивирование с паролем**

**1. Теория.**

**Архиваторы – это программы для создания архивов.** Архивы предназначены для хранения данных в удобном компактном виде. В качестве данных обычно выступают файлы и папки. Как правило, данные предварительно подвергаются процедуре сжатия или упаковки. Поэтому почти каждый архиватор одновременно является программой для сжатия данных. С другой стороны, любая программа для сжатия данных может рассматриваться как архиватор. Эффективность сжатия является важнейшей характеристикой архиваторов. От нее зависит размер создаваемых архивов. Чем меньше архив, тем меньше места требуется для его хранения. Для передачи нужна меньшая пропускная способность канала передачи или затрачивается меньшее время. Преимущества архивов очевидны, если учесть, что данные уменьшаются в размере и в 2 раза, и в 5 раз.

Сжатие данных используется очень широко. Можно сказать, почти везде. Например, документы PDF , как правило, содержат сжатую информацию. Довольно много исполняемых файлов EXE сжаты специальными упаковщиками. Всевозможные мультимедийные файлы ( GIF , JPG , MP 3, MPG ) являются своеобразными архивами.

Основным недостатком архивов является невозможность прямого доступа к данным. Их сначала необходимо извлечь из архива или распаковать. Операция распаковки, впрочем, как и упаковки, требует некоторых системных ресурсов. Это не мгновенная операция. Поэтому архивы в основном применяют со сравнительно редко используемыми данными. Например, для хранения резервных копий или установочных файлов.

В данный момент существует много архиваторов. Они имеют разную распространенность и эффективность. Некоторые интересные архиваторы не известны широкому кругу потенциальных пользователей. Особый интерес представляют оценка и сравнение эффективности сжатия популярных архиваторов.

#### Методы сжатия архиваторов.

Разработано большое количество разнообразных методов, их модификаций и подвидов для сжатия данных. Современные архиваторы, как правило, одновременно используют несколько методов одновременно. Можно выделить некоторые основные.

Кодирование длин серий ( RLE - сокращение от run - length encoding - кодирование длин серий).

Очень простой метод. Последовательная серия одинаковых элементов данных заменяется на два символа: элемент и число его повторений. Широко используется как дополнительный, так и промежуточный метод. В качестве самостоятельного метода применяется, например, в графическом формате BMP .

Словарный метод ( LZ - сокращение от Lempel Ziv - имена авторов).

Наиболее распространенный метод. Используется словарь, состоящий из последовательностей данных или слов. При сжатии эти слова заменяются на их коды из словаря. В наиболее распространенном варианте реализации в качестве словаря выступает сам исходный блок данных.

Основным параметром словарного метода является размер словаря. Чем больше словарь, тем больше эффективность. Однако для неоднородных данных чрезмерно большой размер может быть вреден, так как при резком изменении типа данных словарь будет заполнен неактуальными словами. Для эффективной работы данного метода при сжатии требуется дополнительная память. Приблизительно на порядок больше, чем нужно для исходных данных словаря. Существенным преимуществом словарного метода является простая и быстрая процедура распаковки. Дополнительная память при этом не требуется. Такая особенность особенно важна, если необходим оперативный доступ к данным.

Энтропийный метод (Huffman - кодирование Хаффмена, Arithmetic coding - арифметическое кодирование)

В этом методе элементы данных, которые встречаются чаще, кодируются при сжатии более коротким кодом, а более редкие элементы данных кодируются более длинным кодом. За счет того, что коротких кодов значительно больше, общий размер получается меньше исходного.

Широко используется как дополнительный метод. В качестве самостоятельного метода применяется, например, в графическом формате JPG .

Метод контекстного моделирования ( CM - сокращение от context modeling - контекстное моделирование)

В этом методе строится модель исходных данных. При сжатии очередного элемента данных эта модель выдает свое предсказание или вероятность. Согласно этой вероятности, элемент данных кодируется энтропийным методом. Чем точнее модель будет соответствовать исходным данным, тем точнее она будет выдавать предсказания, и тем короче будут кодироваться элементы данных.

Для построения эффективной модели требуется много памяти. При распаковке приходится строить точно такую же модель. Поэтому скорость и требования к объему оперативной памяти для упаковки и распаковки почти одинаковы. В данный момент методы контекстного моделирования позволяют получить наилучшую степень сжатия, но отличаются чрезвычайно низкой скоростью.

PPM ( PPM - Prediction by Partial Matching - предсказание по частичному совпадению).

Это особый подвид контекстного моделирования. Предсказание выполняется на основании определенного количества предыдущих элементов данных. Основным параметром является порядок модели, который задает это количество элементов. Чем больше порядок модели, тем выше степень сжатия, но требуется больше оперативной памяти для хранения данных модели. Если оперативной памяти недостаточно, то такая модель с большим порядком показывает низкие результаты. Метод PPM особенно эффективен для сжатия текстовых данных.

Предварительные преобразования или фильтрация.

Данные методы служат не для сжатия, а для представления информации в удобном для дальнейшего сжатия виде. Например, для несжатых мультимедиа данных характерны плавные изменения уровня сигнала. Поэтому для них применяют дельта-преобразование, когда вместо абсолютного значения берется относительное. Существуют фильтры для текста, исполняемых файлов, баз данных и другие.

Метод сортировки блока данных ( BWT - сокращение от Burrows Wheeler Transform - по имени авторов).

Это особый вид или группа преобразований, в основе которых лежит сортировка. Такому преобразованию можно подвергать почти любые данные. Сортировка производится над блоками, поэтому данные предварительно разбиваются на части. Основным параметром является размер блока, который подвергается сортировке. Для распаковки данных необходимо проделать почти те же действия, что и при упаковке. Поэтому скорость и требования к оперативной памяти почти одинаковы. Архиваторы, которые используют данный метод, обычно показывают высокую скорость и степень сжатия для текстовых данных.

Непрерывные блоки или непрерывный режим ( Solid mode - непрерывный режим).

Во многих методах сжатия начальный участок данных или файла кодируется плохо. Например, в словарном методе словарь пуст. В методе контекстного моделирования модель не построена. Когда количество файлов большое, а их размер маленький, общая степень сжатия значительно ухудшается за счет этих начальных участков. Чтобы этого не происходило при переходе на следующий файл, используется информация, полученная исходя из предыдущих файлов. Аналогичного эффекта можно добиться простым представлением исходных файлов в виде одного непрерывного файла.

Этот метод используется во многих архиваторах и имеет существенный недостаток. Для распаковки произвольного файла необходимо распаковать и файлы, которые оказались в начале архива. Это необходимо для правильного заполнения словаря или построения модели. Существует и промежуточный вариант, когда используются непрерывные блоки фиксированного размера. Потери сжатия получаются минимальными, но для извлечения одного файла, который находится в конце большого архива, необходимо распаковать только один непрерывный блок, а не весь архив.

#### Сегментирование.

Во всех методах сжатия при изменении типа данных собственно сам переход кодируется очень плохо. Словарь становится не актуальным, модель настроена на другие данные. В этих случаях применяется сегментирование. Это предварительная разбивка на однородные части. Затем эти части кодируются по отдельности или группами.

Особо хочется подчеркнуть, что существует большое количество методов сжатия. Каждый метод обычно ориентирован на один вид или группу реальных данных. Хорошие результаты показывает комплексное использование методов.

Особенности данных

Степень сжатия в основном зависит от исходных данных. Хорошо сжимаются почти все предварительно несжатые данные, например, исполняемые файлы (EXE), тексты (TXT , DOC), базы данных (DBF ), простые несжатые изображения ( BMP ). Ограниченно сжимаются несжатый звук ( WAV ), сложные несжатые изображения ( BMP

). Не сжимаются почти все уже сжатые данные, например, архивы ( ZIP , CAB ), сжатые документы ( PDF ), сжатая графика и видео ( JPG , GIF , AVI , MPG ), сжатый звук ( MP 3). Их сжатие находится в пределах пары процентов за счет служебных блоков и небольшой избыточности.

Для сжатия некоторых специфических данных (текст, несжатые изображения, несжатый звук) существуют специальные методы и архиваторы. Такие архиваторы обеспечивают высокую степень сжатия и высокую скорость. Однако так называемые универсальные архиваторы постепенно дополняются подобными методами. В данный момент только для несжатого звука существуют высокоэффективные специальные архиваторы, такие, как OptimFROG , Monkey Audio . Для текстов и изображений лучшие универсальные архиваторы показывают лучшую степень сжатия. Например, архив изображений получится меньше, если использовать формат BMP и архиватор WinRK вместо специализированных графических форматов, таких как JPEG 2000 ( LossLess - сжатие без потерь).

Большое количество типов данных уже являются сжатыми. Использование архиваторов дает мизерное уменьшение размера. Тем не менее даже в таких случаях эффективное сжатие теоретически возможно. Это обусловлено тем, что в большинстве распространенных форматов файлов, использующих сжатие, применены не самые эффективные методы. Например, в основе формата JPG лежит энтропийное сжатие, которое используется после преобразований Фурье. Данные кодируются неоптимальными блоками, что обусловлено желанием сделать формат JPG устойчивым к повреждениям и возможности частичного извлечения информации. Перекодировав файлы JPG при помощи высокоэффективных методов, можно добиться сжатия порядка 75% от исходного файла (архиватор StuffIt ). Собственно сам исходный файл JPG сжимается обычными архиваторами только до 96%. Однако подобные манипуляции с файлами JPG стали возможны только недавно и еще не получили распространения. В большинстве случаев сжимать уже сжатые данные бесполезно.

Следует различать собственно программу-архиватор, формат архивов и методы сжатия. Даже один и тот же метод сжатия может иметь варианты реализации. Например, существует более десятка программ-архиваторов, которые могут создавать архивы в формате ZIP. В свою очередь данные в формате ZIP могут быть сжаты различными методами: Deflate, Deflate64, BZip2. Метод Deflate имеет несколько реализаций с разной скоростью и степенью сжатия (разница порядка 5%). С помощью этого метода архиватор 7-zip позволяет создавать архивы в формате ZIP и 7Z.

Обычно архиваторы могут создавать архивы в собственном эксклюзивном формате с использованием своих оригинальных методов. Например, архиватор RAR позволяет создавать архивы RAR. В формате архива и методах сжатия заключаются основные преимущества того или иного архиватора.

В простейшем случае архиватор позволяет только упаковать или распаковать один файл. Кроме собственно сжатия данных, современные архиваторы обеспечивают некоторые дополнительные функции. Можно выделить несколько основных:

* сжатие некоторых файлов и целых директорий;
* создание самораспаковывающихся (SFX) архивов. То есть для распаковки архива программа-архиватор не требуется;
* изменение содержимого архива; шифрование содержимого архива;
* информация для восстановления архива при частичном повреждении и возможность восстановления поврежденных архивов;
* разбивка архива на несколько частей или томов;
* консольная версия программы для работы из командной строки; графическая (GUI) версия программы.

Стоит отметить, что, несмотря на формальное наличие, реализация каждой дополнительной функции может быть выполнена на совершенно разном уровне.

Кроме различий в функциональности, можно разбить архиваторы на две группы: асимметричные и симметричные. Асимметричные архиваторы требуют для операции распаковки значительно меньше времени и оперативной памяти, чем для операции упаковки. Это позволяет быстро получать содержимое архива на маломощных компьютерах. Симметричные архиваторы требуют для операций упаковки и распаковки одинаковое время и объем оперативной памяти. Использование таких архиваторов на широком парке компьютеров или для оперативного доступа к содержимому архива ограничено. Известный архиватор RAR в качестве основного использует асимметричный словарный метод сжатия, а для текстов может использовать симметричный PPM-метод. Таким образом, распаковка архивов RAR, сжатых с максимальной степенью сжатия, может быть невозможна на компьютерах с ограниченным объемом оперативной памяти. Все или почти все передовые архиваторы с высокой степенью сжатия являются симметричными.

Точной статистики по распространенности архиваторов у меня нет. Я выскажу свою субъективную точку зрения на основе личного опыта. Безусловно, самым распространенным архиватором являются ZIP и его модификации. По своей распространенности он значительно превосходит ближайших конкурентов. Следом идут RAR и ACE. В последние годы встречается архиватор 7-zip. Других архиваторов и архивов лично мы не встречали. Исключение составляют некогда популярные ARJ и LHA. В данный момент они не актуальны из-за очень низкой степени сжатия.

Несмотря на очень скромные данные о распространенности архиваторов, их существует большое множество. Основная масса относится к категории экспериментальных и архиваторов с ограниченной функциональностью. Тем не менее, каждый их них позволяет выполнять собственно процедуру сжатия данных. Меньшая распространенность увеличивает вероятность ошибок в программе.

### Практика.

* 1. Необходимо создать текстовый файл, содержащий фамилию, имя, отчество студента в объеме 50 записей. Провести архивирование файла. Любым редактором внести изменения согласно задания. В отчете отразить: контрольную сумму исходного файла, сжатого файла, выдаваемые сообщения об ошибках при разархивировании искаженного файла.
  2. Провести архивацию файла с паролем. Внести искажения, попробовать разархивировать. В отчете отразить: контрольную сумму исходного файла, сжатого файла, выдаваемые сообщения об ошибках при разархивировании искаженного файла.
  3. Провести архивацию файла с паролем, состоящим из 3-х цифр. Провести попытку подбора пароля с использованием программного обеспечения. В отчете отразить: контрольную сумму исходного файла, сжатого файла, выдаваемые сообщения, время подбора.

Варианты:

1. архиватор zip. Искажение двух байт.
2. архиватор arj. Искажение трех байт.
3. архиватор rar. Искажение трех байт.
4. архиватор zip. Удаление двух байт.
5. архиватор arj. Удаление трех байт.
6. архиватор rar. Удаление трех байт.
7. архиватор arj. Добавление трех байт.
8. архиватор rar. Добавление трех байт.
9. архиватор zip. Добавление двух байт.
10. архиватор zip. Удаление двух байт.