

многократно отработать технологию компьютерного решения задач, но и создать готовые к внедрению программные продукты.

4. Перечисленные возможности метода проектов особенно важны при подготовке будущих преподавателей, что обуславливает необходимость его использования в педагогическом вузе, особенно при подготовке преподавателей в области современных информационных технологий.

Я. А. Рублев

ФИ УГПУ, гр. ИС-366

ИССЛЕДОВАНИЕ ГРАФИЧЕСКИХ ФОРМАТОВ ФАЙЛОВ

При наличии большой графической базы данных, возникает проблема поиска конкретного файла или изображения по образцу, в связи с этим необходимо исследование и осознание структуры форматов файлов.

Я предлагаю вашему вниманию исследование формата файла на примере формата GIF.

Формат GIF использует алгоритм сжатия LZW. LZW - это способ сжатия данных, который извлекает преимущества при повторяющихся цепочках данных. Поскольку растровые данные обычно содержат довольно много таких повторений, LZW является хорошим методом для их сжатия и раскрытия.

При сжатии LZW манипулирует тремя объектами: строкой символов, таблицей кодов и строкой добавочных символов. По завершении процесса кодирования результатом будет являться конечный код.

Процесс сжатия происходит следующим образом:

1. В начале происходит инициализация строки символов, т.е. каждому отдельному символу, отличному друг от друга, сопоставляется свой код.

2. Берется первый символ из строки символов, т.к. известно, что этот символ присутствует в строке то, в основной код ничего не передается. После этого нашему первому символу в пару добавляется последующий близлежащий

символ, т.е. второй. Эта пара автоматически будет определяться как один символ. Если этот «парный символ» не встречается в строке, то он добавляется в строку добавочных символов и ему сопоставляется первый свободный код. В конечный код переносится код первого символа пары.

3. Берется следующий символ из строки символов, и к нему также добавляем следующий за ним символ. Т.к. строка добавочных символов уже не пуста, то поиск пары будет производиться как по основной так и по добавочной строке символов. Если пара найдет своего «двойника» то в основной код добавится код последнего и данной паре прибавится следующий символ из основной строки и пара превратится в тройной символ, который будет занесен в добавочную строку. В дальнейшем весь процесс кодирования проходит с применением тройных символов. Если нет, то пара заносится в добавочную строку, а в основной код - код первого символа из пары и дальше этот процесс повторяется.

Как только исходная строка символов заканчивается кодировка прекращается.

Пример:

Давайте предположим, что мы имеем 4-символьный алфавит: А,В,С,Д. Поток символов выглядит как АВАСАВА. Давайте сожмем его. Сначала мы инициализируем нашу таблицу цепочек: #0=А, #1=В, #2=С, #3=Д. Первый символ есть А, который входит в таблицу цепочек, следовательно [.с.] становится равным А. Далее мы берем АВ, которая не входит в таблицу, следовательно мы выводим код #0 (для [.с.]), и добавляем АВ в таблицу цепочек с кодом #4. [.с.] становится равным В. Далее мы берем [.с.]А = ВА, которая не входит в таблицу цепочек, следовательно выводим код #1, и добавляем ВА в таблицу цепочек с кодом #5. [.с.] становится равным А. Далее мы берем АС, которая не входит в таблицу цепочек. Выводим код #0, и добавляем АС в таблицу цепочек с кодом #6. Теперь [.с.] равно С. Далее мы берем [.с.]А = СА, которая не входит в таблицу. Выводим #2 для С, и добавляем СА к таблице под кодом #7. Теперь [.с.]=А. Далее мы берем АВ,

которая ВХОДИТ в таблицу цепочек, следовательно [.с.] становится равным АВ, и мы ищем АВА, которой нет в таблице цепочек, поэтому мы выводим код для АВ, который равен #4, и добавляем АВА в таблицу цепочек под кодом #8. [.с.] равно А. Мы не можем более взять символов, поэтому мы выводим код #0 для А и заканчиваем. Следовательно, поток кодов равен #0#1#0#2#4#0.

Пример:

Пусть мы сжимаем последовательность 45, 55, 55, 151, 55, 55, 55. Тогда, согласно изложенному выше алгоритму, мы поместим в выходной поток сначала код очистки <256>, потом добавим к изначально пустой строке “45” и проверим, есть ли строка “45” в таблице. Поскольку мы при инициализации занесли в таблицу все строки из одного символа, то строка “45” есть в таблице. Далее мы читаем следующий символ 55 из входного потока и проверяем, есть ли строка “45, 55” в таблице. Такой строки в таблице пока нет. Мы заносим в таблицу строку “45, 55” (с первым свободным кодом 258) и записываем в поток код <45>. Можно коротко представить архивацию так:

“45” — есть в таблице;

“45, 55” — нет. Добавляем в таблицу <258>“45, 55”. В поток: <45>;

“55, 55” — нет. В таблицу: <259>“55, 55”. В поток: <55>;

“55, 151” — нет. В таблицу: <260>“55, 151”. В поток: <55>;

“151, 55” — нет. В таблицу: <261>“151, 55”. В поток: <151>;

“55, 55” — есть в таблице;

“55, 55, 55” — нет. В таблицу: “55, 55, 55” <262>. В поток: <259>;

Последовательность кодов для данного примера, попадающих в выходной поток: <256>, <45>, <55>, <55>, <151>, <259>.

Процесс сжатия выглядит достаточно просто. Мы считываем последовательно символы строки и проверяем, есть ли в строке такой символ. Если символ есть, то мы считываем следующий символ, а если нет, то мы

вносим в конечный код - код для предыдущего символа, вносим строку в строку добавочных символов и начинаем поиск снова.

В дальнейшем конечный код переводится в шестнадцатеричный код и уже в файле отображается в ASCII кодировке. Дальнейшая задача состоит в том, как при помощи уже определенного кода вычленить именно ту часть, которая необходима для сопоставления с кодом уже имеющегося файла.

Исследование графических форматов имеет важное значение для работы с графическими базами данных. Например, это даже может помочь в раскрытии дела о похищенной памяти из нашего университета. Сопоставление отпечатков пальцев произойдет намного быстрее и легче с применением кодировки в числовом формате. А также использование структуры форматов может помочь в дистанционном обучении.

В. Ю. Турушкина
матем. факультет УрГПУ

ПРИМЕНЕНИЕ КОМПЬЮТЕРА ПРИ ИЗУЧЕНИИ ОСНОВ ГЕОМЕТРИИ В 5-6 КЛАССЕ

Анализ существующих ППС по геометрии позволил разделить их на несколько групп: иллюстрирующие, моделирующие, контролирующие. Содержательная часть каждой из рассмотренных программ отражает достаточно небольшой раздел школьного курса геометрии, что позволяет использовать данное ППС на ограниченном числе уроков.

Автором предпринята попытка создать ППС, глубоко интегрированное в учебный процесс, учитывающее многообразие школьных программ, содержащее значительный объем теоретического материала.

Общая информация о программе “Геометрия”.

Предлагаемое ППС позволяет организовать групповые и (или) индивидуальные занятия по изучению основ геометрии в 5-6 классе, с последующим контролем качества усвоения материала.