

## Практическая работа

### Алгоритм RLE

1. Используя алгоритм RLE, закодируйте последовательность символов

**BBBBBACCCABBBBB**

Запишите результат в виде шестнадцатеричных кодов (каждый символ кодируется в виде байта, который представлен двумя шестнадцатеричными цифрами). Проверьте полученный результат с помощью программы RLE.

Ответ:

2. Раскодируйте последовательность, упакованную с помощью алгоритма RLE (приводятся шестнадцатеричные коды): 01 4D 8E 41 01 4D 8E 41<sub>16</sub>. Для определения символов по их шестнадцатеричным кодам используйте таблицу ASCII. В приведённой таблице в первом столбце записана первая цифра шестнадцатеричного кода символа, а в первой строке – вторая. Например, символ «&» имеет шестнадцатеричный код 26<sub>16</sub>.

	.0	.1	.2	.3	.4	.5	.6	.7	.8	.9	.A	.B	.C	.D	.E	.F
0.	NU L	SO H	ST X	ET X	EO T	EN Q	AC K	BE L	BS	TA B	LF	VT	FF	C R	S O	SI
1.	DL E	DC 1	DC 2	DC 3	DC 4	NA K	SY N	ET B	CA N	E M	SU B	ES C	FS	G S	R S	US
2.		!	"	#	\$	%	&	'	(	)	*	+	,	—	.	/
3.	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4.	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5.	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6.	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7.	p	q	r	s	t	u	v	w	x	y	z	{		}	~	DEL

Ответ:

3. Определите количество байтов в исходной и распакованной последовательности (см. предыдущее задание) и вычислите коэффициент сжатия:

Сжатая последовательность	Несжатая последовательность	Коэффициент сжатия

4. Проверьте результат, полученный в предыдущем пункте, с помощью программы RLE. Предложите два способа проверки.
5. Постройте последовательности, которые сжимаются алгоритмом RLE ровно в 2 раза, в 4 раза, в 5 раз. Проверьте свои ответы с помощью программы RLE.

Несжатая последовательность	Сжатая последовательность	Коэффициент сжатия
AAAA		
AAAAAAAAA		
AAAAAAAAAAA		

6. Придумайте три последовательности, которые невозможно сжать с помощью алгоритма RLE:

Несжатая последовательность	«Сжатая» последовательность	Коэффициент сжатия
M		
MA		
POP		

7. Используя программу RLE, примените RLE-сжатие к следующим файлам и найдите для каждого из них коэффициент сжатия:

Файл	Размер без сжатия	Размер после сжатия	Коэффициент сжатия
grad_vert.bmp			
grad_horz.bmp			
grad_diag.jpg			

8. Объясните результаты, полученные в предыдущем пункте:

- почему не удастся сжать рисунки в формате JPEG?

*Ответ:*

- почему для двух рисунков в формате BMP одинакового размера коэффициенты сжатия по алгоритму RLE так сильно отличаются?  
*Подсказка:* откройте эти рисунки в любой программе просмотра.

*Ответ:*

9. Оцените максимально достижимый коэффициент сжатия с помощью рассмотренного в учебнике варианта RLE-алгоритма. В каком случае его удастся достичь?

*Ответ:*

10. Оцените коэффициент сжатия с помощью RLE-алгоритма в худшем случае. Опишите этот худший случай.

*Ответ:*

## Сравнение алгоритмов сжатия

При выполнении этой работы используются программы **RLE** (алгоритм сжатия RLE) и **Huffman** (кодирование Хаффмана и Шеннона-Фано).

1. Запустите программу **Huffman.exe** и закодируйте строку «БЫЛИ У БАБУСИ ДВА ВЕСЕЛЫХ ГУСЯ», используя методы Шеннона-Фано и Хаффмана. Запишите результаты в таблицу:

	Шеннон и Фано	Хаффман
Длина основного кода		
Длина кодовой таблицы (дерева)		
Коэффициент сжатия (по основным кодам)		
Коэффициент сжатия (с учетом дерева кодов)		

Сделайте выводы.

*Ответ:*

Как, по вашему мнению, будет изменяться коэффициент сжатия при увеличении длины текста, при условии, что набор символов и частота их встречаемости останутся неизменной? Проверьте ваш вывод с помощью программы (например, можно несколько раз скопировать ту же фразу).

*Ответ:*

2. Используя кнопку *Анализ файла* в программе **Huffman**, определите предельный теоретический коэффициент сжатия для файла **a.txt**<sup>1</sup> при побайтном кодировании.

*Ответ:*

3. С помощью программ **RLE** и **Huffman** выполните сжатие файла **a.txt** разными способами. Запишите результаты в таблицу:

	RLE	Шеннон и Фано	Хаффман
Размер сжатого файла			
Коэффициент сжатия			

Объясните результат, полученный с помощью алгоритма RLE.

*Ответ:*

<sup>1</sup> Этот файл имеет объем 1 Мбайт и состоит из одних символов «А».

4. Используя кнопку *Анализ файла* в программе **Huffman**, определите предельный теоретический коэффициент сжатия для файла **a.txt.huf** при побайтном кодировании. Объясните результат.

*Ответ:*

5. Примените несколько раз повторное сжатие этого файла с помощью алгоритма Хаффмана (новые файлы получают имена **a.txt.huf2**, **a.txt.huf3** и т.д.) и заполните таблицу, каждый раз выполняя анализ полученного файла.

	Размер файла	Предельный коэффициент сжатия
<b>a.txt</b>		
<b>a.txt.huf</b>		
<b>a.txt.huf2</b>		
<b>a.txt.huf3</b>		
<b>a.txt.huf4</b>		
<b>a.txt.huf5</b>		
<b>a.txt.huf6</b>		

Объясните, почему с некоторого момента при повторном сжатии файла его размер увеличивается.

*Ответ:*

6. Выполните те же действия, используя метод Шеннона-Фано.

	Размер файла	Предельный коэффициент сжатия
<b>a.txt</b>		
<b>a.txt.shf</b>		
<b>a.txt.shf2</b>		
<b>a.txt.shf3</b>		
<b>a.txt.shf4</b>		
<b>a.txt.shf5</b>		
<b>a.txt.shf6</b>		

Объясните, почему с некоторого момента при повторном сжатии файла его размер увеличивается.

*Ответ:*

7. Сравните результаты сжатия этого файла с помощью алгоритма RLE, лучшие результаты, полученные методами Шеннона-Фано и Хаффмана, а также результат сжатия этого файла каким-нибудь архиватором.

Размер файла	Предельный коэффициент
--------------	------------------------

		<b>сжатия</b>
<b>RLE</b>		
<b>Хаффман</b>		
<b>Шеннон и Фано</b>		
<b>ZIP</b>		
<b>RAR</b>		
<b>7Z</b>		

Объясните результаты и сделайте выводы.

*Ответ:*