##### Задание 10

1. Файл data.bin содержит набор 32-битных целых чисел со знаком. Размер файла не превышает 1MiB.

Вывести одно целое число – сумму чисел из файла data.bin.

2. Файл data.bin содержит набор данных, состоящих из строки (40 символов, включая символ окончания строки) и 32-битного целого числа.

Первая строка ввода содержит одно целое число *N* (1 ≤ *N* ≤ 250000) – количество запросов. Далее следует *N* строк, каждая строка содержит один из запросов.

? *x*  – вывести строку и число из записи *x*;

+ *s* *a* – добавить новую запись под номером *K* со строкой *s* и числом *a*;

= *x* *s* *a* – в запись *x* записать строку *s* и число *a*.

Здесь *x*  – номер от 0 до *K*−1, *K* – текущее количество записей, *s* – строка из строчных латинских букв (длиной от 1 до 39), *a* – целое число.

3. Дан файл archive.zip, вывести список файлов в zip-архиве.

Последние 22 байта zip-архива содержат следующую информацию:

|  |  |  |
| --- | --- | --- |
| Смещение | Размер | Описание |
| 0 | 4 | 0x06054b50 |
| 4 | 2 | - |
| 6 | 2 | - |
| 8 | 2 | - |
| 10 | 2 | Количество файлов в архиве |
| 12 | 4 | - |
| 16 | 4 | Смещение от начала архива до списка файлов |
| 20 | 2 | 0 |

Каждый элемент списка файлов содержит следующую информацию:

|  |  |  |
| --- | --- | --- |
| Смещение | Размер | Описание |
| 0 | 4 | 0x02014b50 |
| 4 | 2 | - |
| 6 | 2 | - |
| 8 | 2 | - |
| 10 | 2 | - |
| 12 | 2 | - |
| 14 | 2 | - |
| 16 | 4 | - |
| 20 | 4 | Размер сжатого файла |
| 24 | 4 | Размер несжатого файла |
| 28 | 2 | Длина имени файла (n) |
| 30 | 2 | Длина дополнительной информации (m) |
| 32 | 2 | Длина комментария (k) |
| 34 | 2 | - |
| 36 | 2 | - |
| 38 | 4 | - |
| 42 | 4 | Смещение от начала архива до сжатого файла |
| 46 | n | Имя файла |
| 46+n | m | Дополнительная информация |
| 46+n+m | k | Комментарий |

Для каждого файла вывести его имя, размер сжатого файла и размер несжатого файла

4. Ввод содержит некоторый текст до 106 символов. Закодировать этот текст с помощью кода Хаффмана.

Дерево кодирования строится следующим образом.

Для каждого символа текста, включая пробелы и переходы на новую строку, подсчитывается количество его появлений в тексте. Для всех символов, появляющихся в тексте хотя бы один раз, добавляется вершина. Кроме того в набор вершин включается нулевой символ для конца текста с количеством появлений 1.

После этого выбирается две вершины с наименьшим количеством появлений и они заменяются на одну вершину с количеством появлений, равным их сумме, а заменяемые вершины становятся ее потомками. Это действие выполняется до тех пор, пока не останется одна вершина.

Первая часть файла содержит структуру дерева кодирования. Бит 1 указывает на лист дерева, далее следует 8 бит ASCII-кода, начиная со старшего. Бит 0 указывает на промежуточный лист дерева, далее следует описание левой ветки дерева, кодируемой битом 0, и правой ветки дерева, кодируемой битом 1.

Вторая часть файла содержит закодированный текст. Коды символов задаются в файле, начиная со старшего бита, и соответствуют пути от корня дерева до листа-символа. Код для нулевого символа завершает вывод закодированного текста. Неполный байт добавляется до полного произвольными битами и выводится.

Вывести в файл output.bin структуру дерева и закодированный текст. Если существует несколько вариантов кодирования, то вывести любой из них.

5. Бинарный файл input.bin содержит текст, закодированный с помощью кода Хаффмана.

Первая часть файла содержит структуру дерева кодирования. Бит 1 указывает на лист дерева, далее следует 8 бит ASCII-кода, начиная со старшего. Бит 0 указывает на промежуточный лист дерева, далее следует описание левой ветки дерева, кодируемой битом 0, и правой ветки дерева, кодируемой битом 1.

Вторая часть файла содержит закодированный текст. Коды символов задаются в файле, начиная со старшего бита, и соответствуют пути от корня дерева до листа-символа. Символ с кодом 0 завершает ввод и не выводится.

Вывести раскодированный текст.