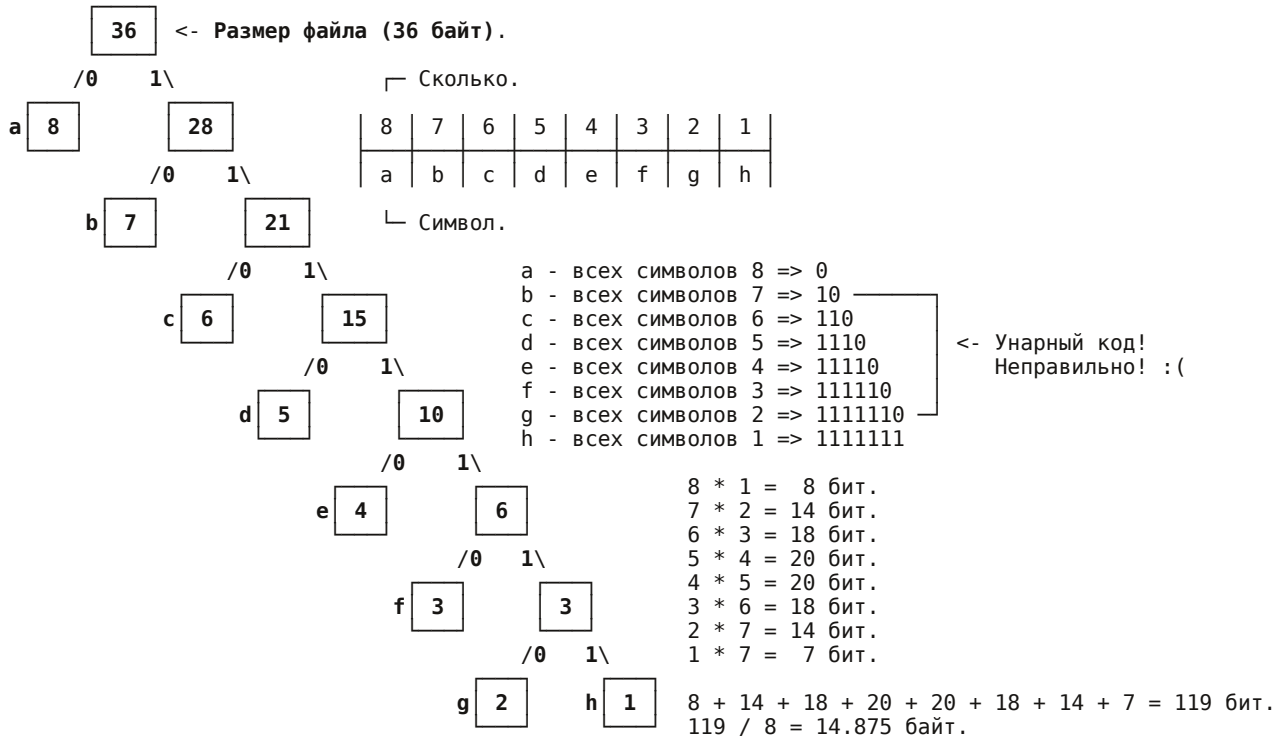


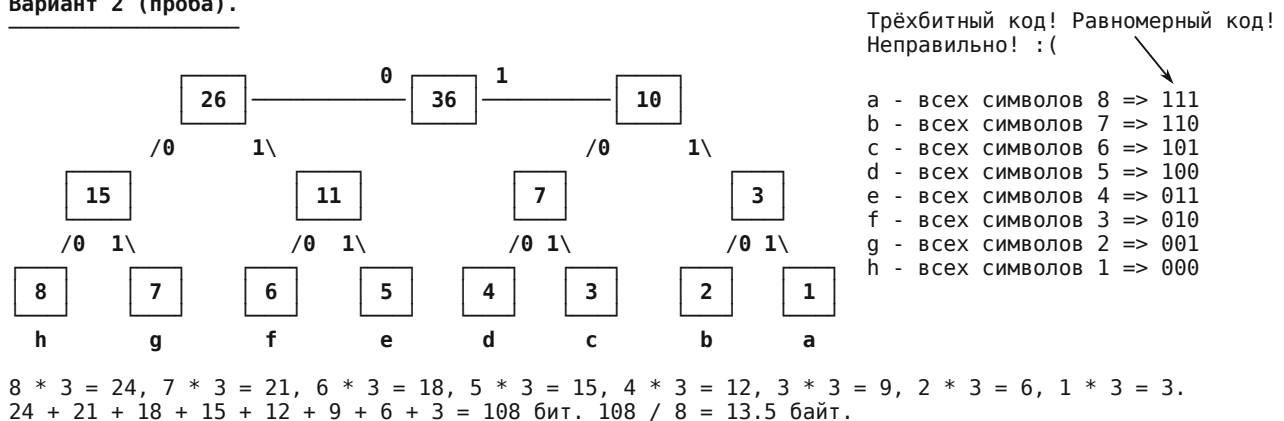
Шеннона-Фано

Книга: Алгоритмы сжатия данных без потерь. Е.Р.Пантелеев, А.Л.Алыкова.
Стр. 48 (Шеннона-Фано, там пример для слова абракадабра).

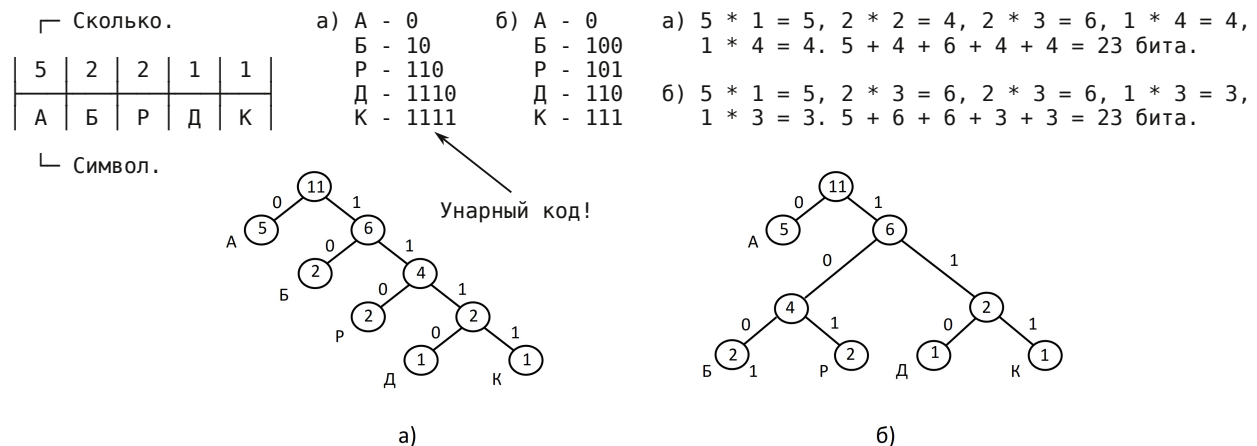
Вариант 1 (проба).



Вариант 2 (проба).



Скан из книги (деревья, см. ниже): Алгоритмы сжатия данных без потерь. Е.Р.Пантелеев, А.Л.Алыкова.
Стр. 48 (Шеннона-Фано, пример для слова абракадабра). Варианты деревьев а и б.



1001 1010 1011 1100 1101 1110 1111 0000 0001

1001 1010 1011 1100 1101 1110 1111 0000 0001

Шеннона-Фано

Если использовать эти веса, то решение совпадает с решением в книге "Алгоритмы сжатия данных без потерь. Е.Р.Пантелеев, А.Л.Алыкова. Стр. 48" слово АБРАКАДАБРА. Также решение совпадает с решением "Собственное творчество 3".

```
//
// Веса собственного производства.
```

```
// JavaScript.
```

```
function ves()
```

```
{
//      А К Р С Н Я (КРАСНАЯКРАСКА)
// var mas = [ 4, 3, 2, 2, 1, 1 ];
```

```
//      А Б Р Д К (АБРАКАДАБРА)
var mas = [ 5, 2, 2, 1, 1 ];
```

```
var xl, xr, z;
var sum_l, sum_r;
```

```
total = mas.length
```

```
xl = 0;
xr = total-1;
```

```
sum_l = mas[xl++];
sum_r = mas[xr--];
```

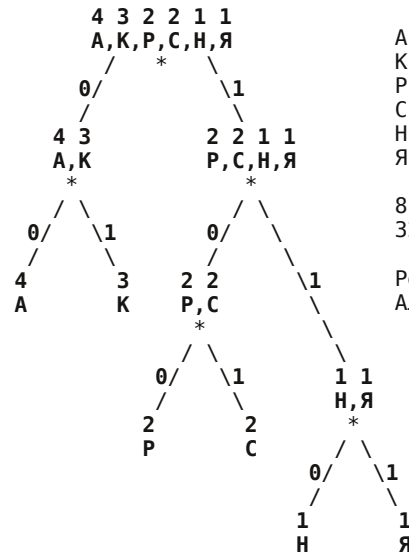
```
for(z = 2; z < total;)
```

```
{
  if (sum_l > sum_r)
  {
    sum_r = sum_r + mas[xr--];
    z++;
  }
  else
  {
    sum_l = sum_l + mas[xl++];
    z++;
  }
}
```

```
alert(sum_l + " " + sum_r + " " + z);
}
```

КРАСНАЯКРАСКА - возвращает: sum_l = 7, sum_r = 6, z = 6.
АБРАКАДАБРА - возвращает: sum_l = 5, sum_r = 6, z = 5.

КРАСНАЯКРАСКА на новый лад (используем собственные веса).
Собственное творчество 4.



A - 00 4 * 2 = 8
K - 01 3 * 2 = 6
P - 100 2 * 3 = 6
C - 101 2 * 3 = 6
H - 110 1 * 3 = 3
Y - 111 1 * 3 = 3

8 + 6 + 6 + 6 + 3 + 3 = 32 бита.
32 / 8 = 4 байта.

Результат совпадает с результатом
Алексея Сергеева.

Короткие последовательности слева, длинные - справа.
A - 00, K - 01 <= короткие. P, C, H, Y <= длинные.

Для заметок: _____
