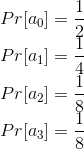
Thus, the alphabet's entropy specifes to within one bit how many bits on the average need to be used to send the alphabet. The smaller an alphabet's entropy, the fewer bits required for digital transmission of fles expressed in that alphabet.

###### Example 6.2

A four-symbol alphabet has the following probabilities.



and an entropy of 1.75 bits (Example 6.1). Let's see if we can fnd a codebook for this four-letter alphabet that satisfes the Source Coding Theorem. The simplest code to try is known as the **simple binary code**: convert the symbol's index into a binary number and use the same number of bits for each symbol by including leading zeros where necessary.



(6.53)

Whenever the number of symbols in the alphabet is a power of two (as in this case), the average number of bits



equals log2*K*, which equals 2 in this case. Because the entropy equals 1.75bits, the simple binary code indeed satisfes the Source Coding Theorem we are within one bit of the entropy limit but you might wonder if you can do better. If we choose a codebook with difering number of bits for the symbols, a smaller average number of bits can indeed be obtained. The idea is to use shorter bit sequences for the symbols that occur more often. One codebook like this is



(6.54)

Now



We can reach the entropy limit! The simple binary code is, in this case, less efcient than the unequal-length code. Using the efcient code, we can transmit the symbolic-valued signal having this alphabet 12.5% faster. Furthermore, we know that no more efcient codebook can be found because of Shannon's Theorem.