Exercise 2.71.

Suppose we have a Huffman tree for an alphabet of n symbols, and that the relative frequencies of the symbols are $1, 2, 4, \ldots, 2^{n-1}$. Sketch the tree for $n=5$; for $n=10$. In such a tree (for general n) how may bits are required to encode the most frequent symbol? the least frequent symbol?

Answer.

Figure 1 and Figure 2 shows the corresponding Huffman tree for $n=5$ and $n=10$ where the relarive

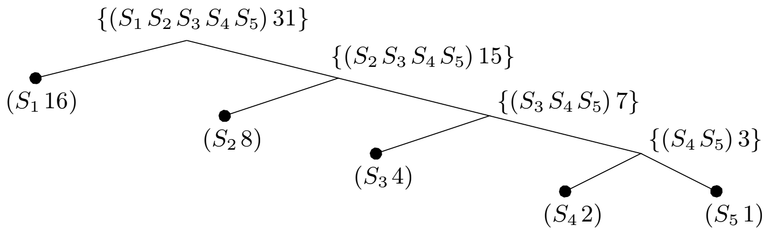

Figure 1. A Huffman tree for $n=5$ where the relarive frequencies of the symbols are $1, 2, 4, \ldots, 2^{n-1}$.

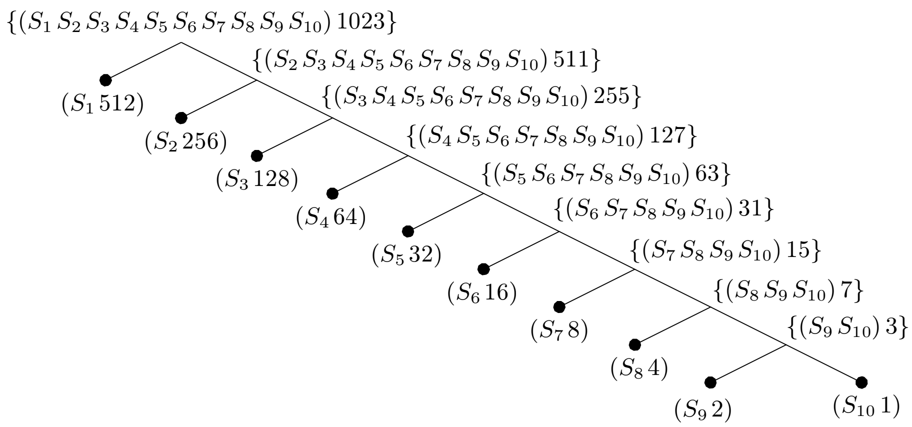

Figure 2. A Huffman tree for $n=10$ where the relarive frequencies of the symbols are $1, 2, 4, \ldots, 2^{n-1}$.

frequencies of the symbols are $1, 2, 4, \ldots, 2^{n-1}$.

We can conclude from these trees that it takes up only 1 bit to encode the most frequent symbol, no matter how many symbols are contained in the alphabet. To encode the least frequent symbol, however, we have to "go down" the tree until we reach the designated least frequent symbol, which is lives in the deepest branch in the tree. Hence we are required to use $n-1$ bits to encode it, which is the same amount of steps we spent in accessing the corresponding node.