✖

Interview Questions: Data Compression

3/3 points earned (100%)

Excellent!

Retake

Course Home

✓ 1 / 1 points

1.

Ternary Huffman codes. Generalize the Huffman algorithm to codewords over the ternary alphabet (0, 1, and 2) instead of the binary alphabet. That is, given a bytestream, find a prefix-free ternary code that uses as few trits (0s, 1s, and 2s) as possible. Prove that it yields optimal prefix-free ternary code.

a

Thank you for your response.

Hint: Combine smallest 3 probabilities at each step (instead of smallest 2). Don't forget to handle the case when the number of symbols is not of the form $3 + 2k$ for some integer k.

✓ 1 / 1 points

2.

Uniquely decodable code.

- Identify an optimal uniquely-decodable code that is neither prefix free nor suffix tree.
- Identify two optimal prefix-free codes for the same input that have a different distribution of codeword lengths.

a

Thank you for your response.

✓ 1 / 1 points

3.

Move-to-front coding. Design an algorithm to implement move-to-front encoding so that each operation takes logarithmic time in the worst case. That is, maintain alphabet of symbols in a list. A symbol is encoded as the number of symbols that precede it in the list. After encoding a symbol, move it to the front of the list.

a

Thank you for your response.