→ Compare and contrast binary Huffman coding and non binary huffman algorithms. When would you use each?

Huffman coding is a lossless data compression technique that assigns variable length prefix codes to symbols based on their probabilities. The standard form is binary huffman coding but the algorithm can be generalized to non-binary (D-ary) Huffman coding, where each node can have more than two children.

Binary Huffman coding:

Binary huffman coding constructs a tree in which every internal node has exactly two children. The edges are labeled 0 and 1 producing binary codewords.

- → Branching Factor = 2
- → Code alphabet = {0,1}
- → Produces binary bitstream
- → Optimal for binary channels
- → Simple implementation
- → Widely used in compression standards

Non-binary huffman coding :

Non binary huffman coding generalizes the method by allowing each internal node to have D children, where D>2 The code symbols are digits from 0 - D-1

Characteristics :

→ Branching Factor = D(>2)

→ Code alphabet = {0, 1, ... D-1}

→ Produces D-ary codewords

→ Shallower tree than binary

→ More complex construction.

eg: ternary(D=3), quaternary(D=4) Huffman codes

In short, Binary huffman is used for binary computers and non binary huffman is used for multi-symbol systems