Design and Analysis of Algorithms
Unit -4

Bharathi R
Department of Computer Science & Engineering

DESIGN AND ANALYSIS OF ALGORITHMS

Unit 4: Greedy Technique- Huffman Trees

Bharathi R

Department of Computer Science & Engineering

Huffman Trees

The Huffman trees are constructed for encoding a given text of n characters.

While encoding a given text, each character is associated with some of bits called the ***codeword***.

- **Fixed length encoding**: Assigns to each character a bit string of the same length.
- **Variable length encoding**: Assigns codewords of different lengths to different characters.
- **Prefix free code**: In Prefix free code, no codeword is a prefix of a codeword of another character.

Binary prefix code :

- The characters are associated with the leaves of a binary tree.
- All left edges are labeled as 0, and right edges as 1.
- Codeword of a character is obtained by recording the labels on the simple path from the root to the character's leaf.
- Since, there is no simple path to a leaf that continues to another leaf, no codeword can be a prefix of another codeword.

About Huffman Algorithm:

- Invented by David A Huffman in 1951.
- Constructs binary prefix code tree.
- Huffman's algorithm achieves data compression by finding the best variable length binary encoding scheme for the symbols that occur in the file to be compressed. Huffman coding uses frequencies of the symbols in the string to build a variable rate prefix code
  - Each symbol is mapped to a binary string
  - More frequent symbols have shorter codes
  - No code is a prefix of another code (prefix free code)
- Huffman Codes for data compression achieves 20-90% Compression.

Huffman Algorithm:

Input: Alphabet and frequency of each symbol in the text.

Step 1: Initialize n one-node trees (forest) and label them with the symbols of the alphabet given. Record the frequency of each symbol in its tree's root to indicate the tree's weight. (Generally, the weight of a tree will be equal to the sum of the frequencies in the tree's leaves.)

Step 2: Repeat the following operation until a single tree is obtained. Find two trees with the smallest weight. Make them the left and right subtree of a new tree and record the sum of their weights in the root of the new tree as its weight.

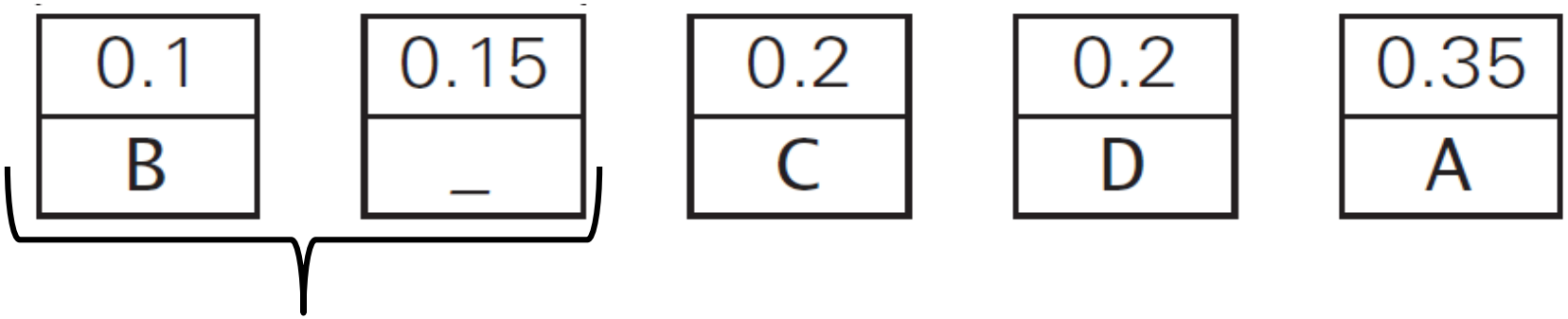
Design and Analysis of Algorithms

Huffman Algorithm:

EXAMPLE

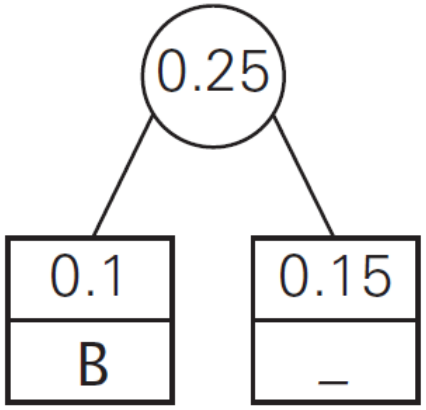
symbol	A	B	C	D	_
frequency	0.35	0.1	0.2	0.2	0.15

1.First arrange the characters in ascending order of their probabilities

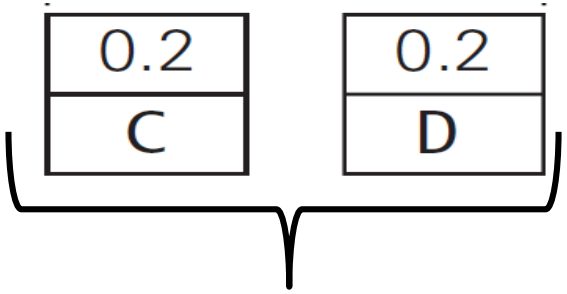


Combine these two nodes and form a single node

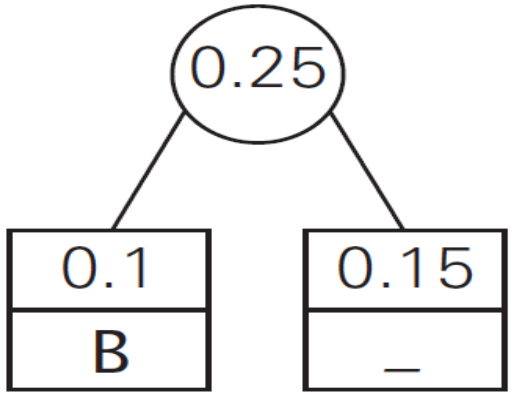
Huffman Algorithm:

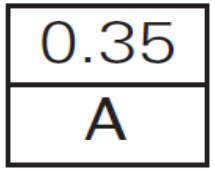


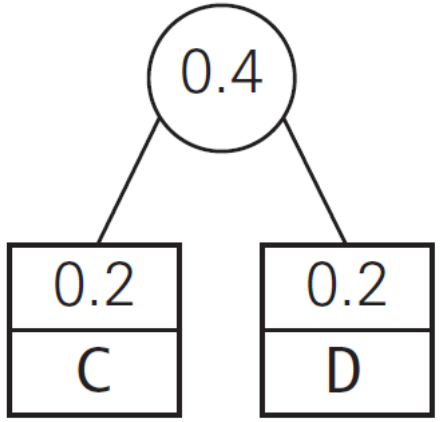
2.Again arrange probabilities in ascending order



Combine these nodes



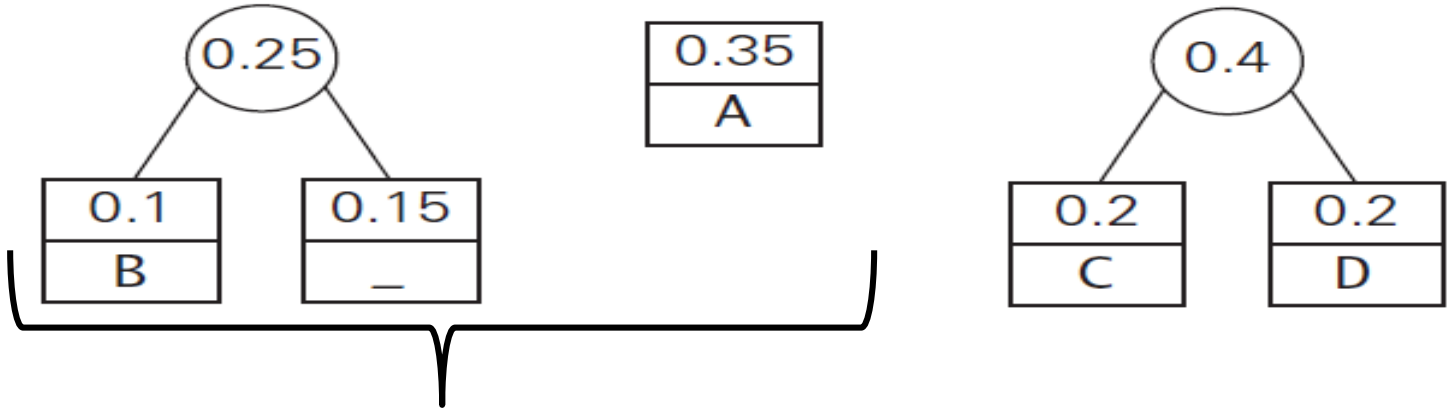




Design and Analysis of Algorithms

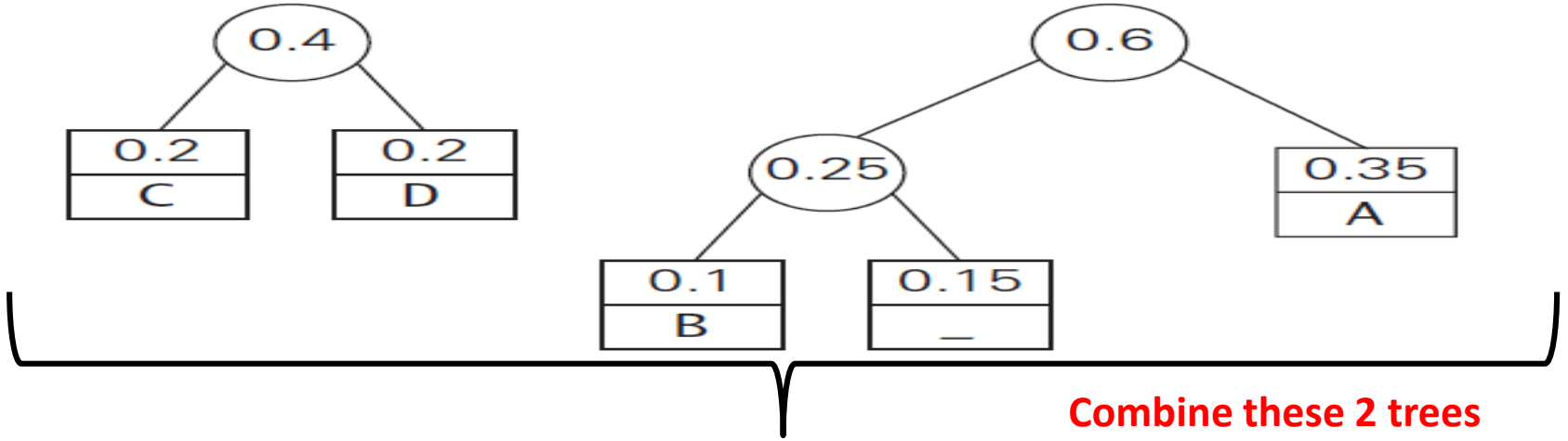
Huffman Algorithm:

3.Again arrange probabilities in ascending order



Combine these 2 trees

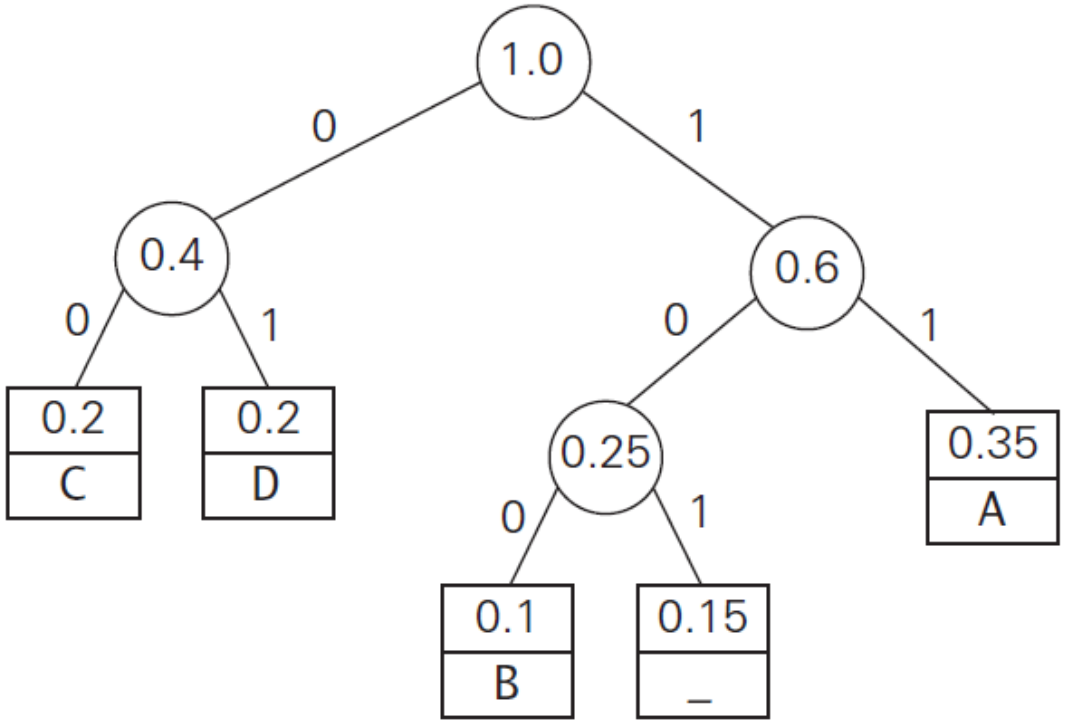
4. Arrange probabilities in ascending order



Combine these 2 trees

Design and Analysis of Algorithms

Huffman Tree



DAD is encoded as 011101, and
10011011011101 is decoded as BAD_AD

The resulting codewords are as follows

symbol	A	B	C	D	_
frequency	0.35	0.1	0.2	0.2	0.15
codeword	11	100	00	01	101

Compression Ratio

Compression Ratio : A standard measure of a compression algorithm's efficiency.

Character	A	B	C	D	-
probability	0.4	0.1	0.2	0.15	0.15

Character	A	B	C	D	-
probability	0.4	0.1	0.2	0.15	0.15
Code word	0	100	111	101	110

Compute compression ratio:

Bits per character = Codeword length * Frequency

= (1 * 0.4) + (3 * 0.1) + (3 * 0.2) + (3 * 0.15) + (3 * 0.15)

= 2.20

Compression ratio is = (3 – 2.20)/ 3 . 100% = 26.6%

Features of Huffman's Encoding

- Huffman's encoding is one of the most important file-compression methods.
- In addition to its simplicity and versatility, it yields an optimal, i.e., minimal-length, encoding (provided the frequencies of symbol occurrences are independent and known in advance).
- The simplest version of Huffman compression calls, in fact, for a preliminary scanning of a given text to count the frequencies of symbol occurrences in it. Then these frequencies are used to construct a Huffman coding tree and encode the text.

Design and Analysis of Algorithms

Text Books

Chapter-9 Greedy Technique
Introduction to the Design & Analysis of Algorithms- Anany Levitin

THANK YOU

Bharathi R

Department of Computer Science & Engineering

rbharathi@pes.edu