HUFFMAN CODING

A Greedy Approach

Badhan Das
Zarin Tasnim Promi

Bangladesh University of Engineering and Technology

December 5, 2015

Traditional Encoding

- Characters are typically encoded by their ASCII codes with 8 bits per character.
- As example the word 'ABC' takes 24 bits to encode,

A Common Scenerio

- Memory Restrictions
- Multiple file sending Restrictions

Solution

- Text Compression by Implementing Huffman Coding

Figure : Text Compression

Huffman Coding

- Variable length representation based on symbol frequency

Huffman Coding

- Variable length representation based on symbol frequency
- Efficient when symbol probabilities vary widely

Huffman Coding

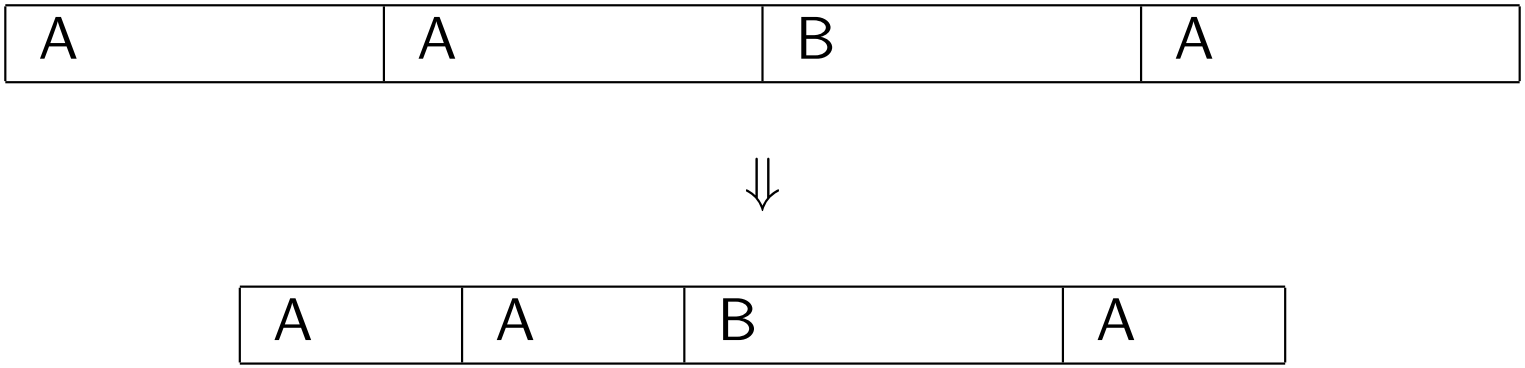
- Variable length representation based on symbol frequency
- Efficient when symbol probabilities vary widely
- Capable of saving 20% to 90% space

Huffman Coding

- Variable length representation based on symbol frequency
- Efficient when symbol probabilities vary widely
- Capable of saving 20% to 90% space
- Uses an optimal encoding tree to determine the code words and vice-versa

Main Idea

- Use fewer bits to represent frequent symbols
- Use more bits to represent infrequent symbols



Huffman Binary tree Construction

Steps:

- Frequency table construction
- Sorting the table
- Adding the lowest two frequency
- Denoting a new symbol and updating the table
- Have to repeat the last 3 process until there is one node
- Performing a traversal of the tree to obtain new code words
- Going left is a 0 going right is a 1

Huffman Encoding

- Encodes high-frequency characters with short code words
- Uses an optimal frequency tree to determine the code words

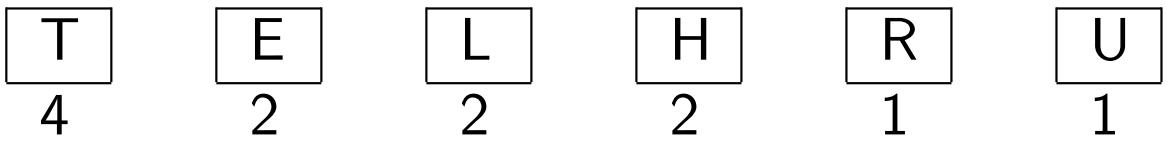
Huffman Encoding Simulation

EXAMPLE : **TELL THE TRUTH**

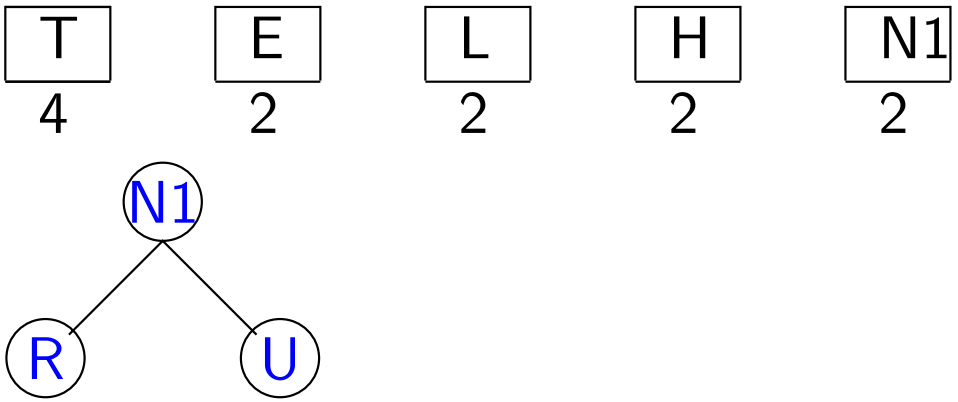
Frequency Table :

Symbol	Frequency
T	4
E	2
H	2
L	2
R	1
U	1

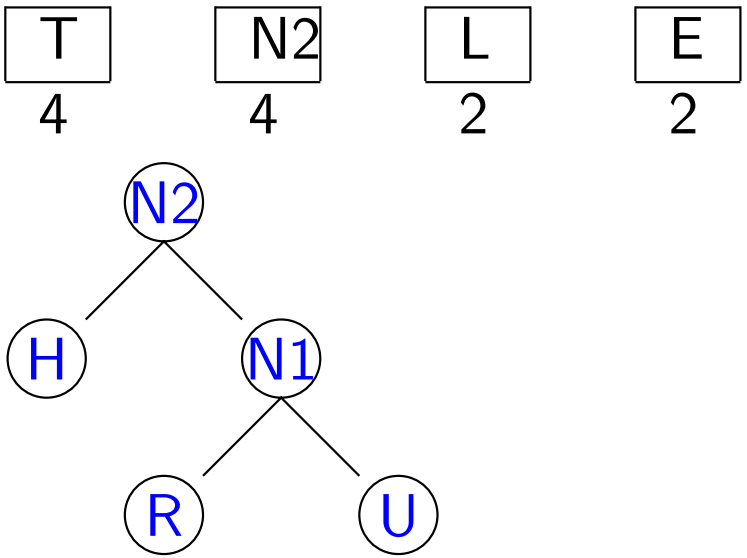
Huffman Encoding Simulation



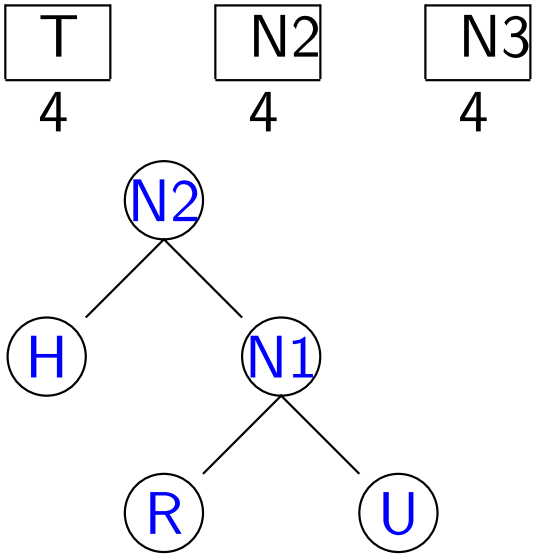
Huffman Encoding Simulation

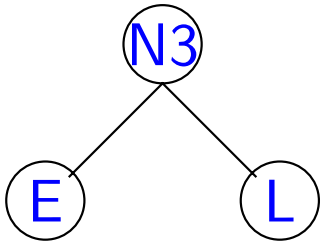


Huffman Encoding Simulation

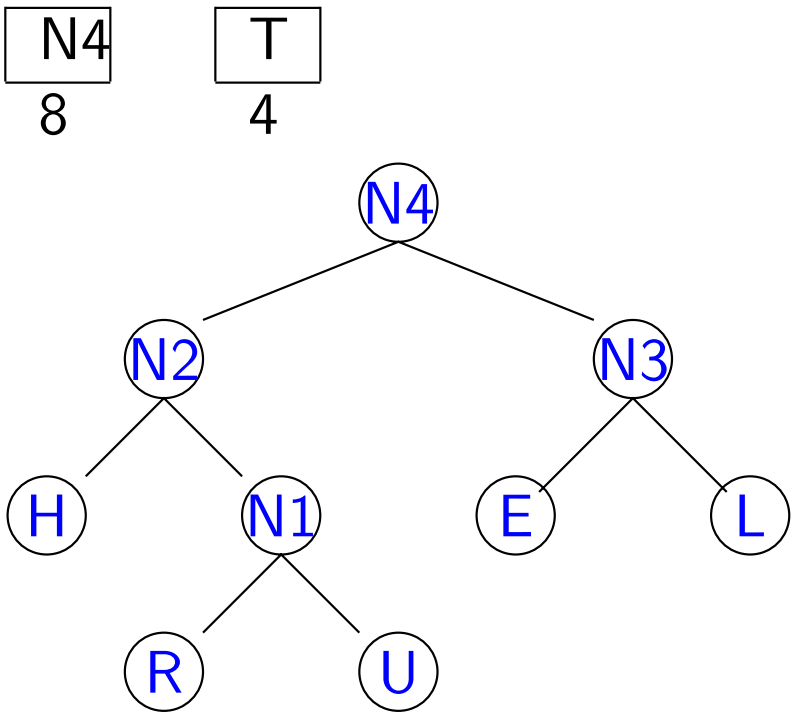


Huffman Encoding Simulation





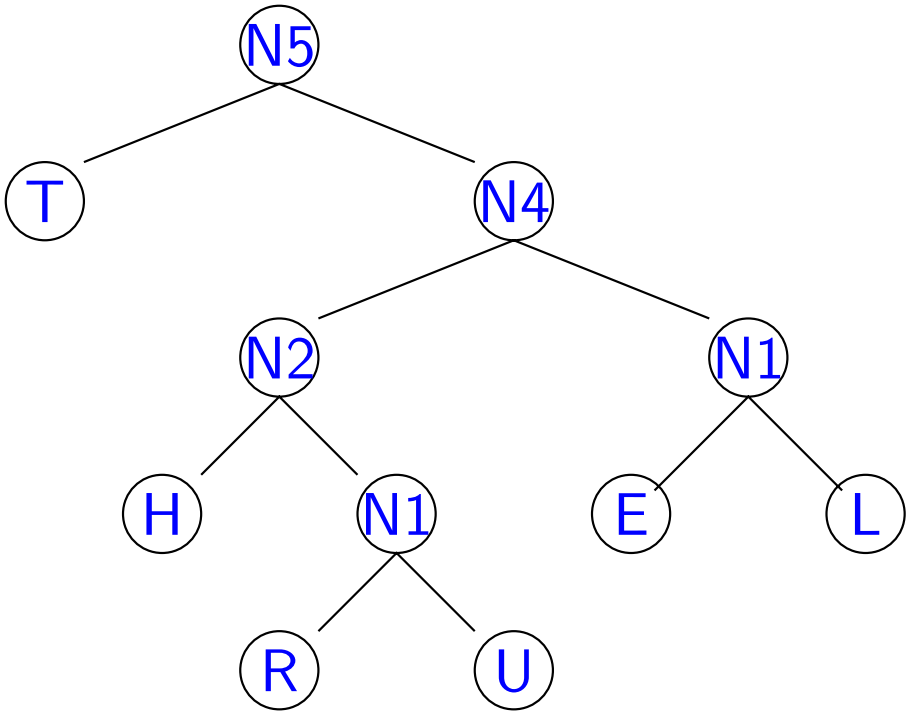
Huffman Encoding Simulation



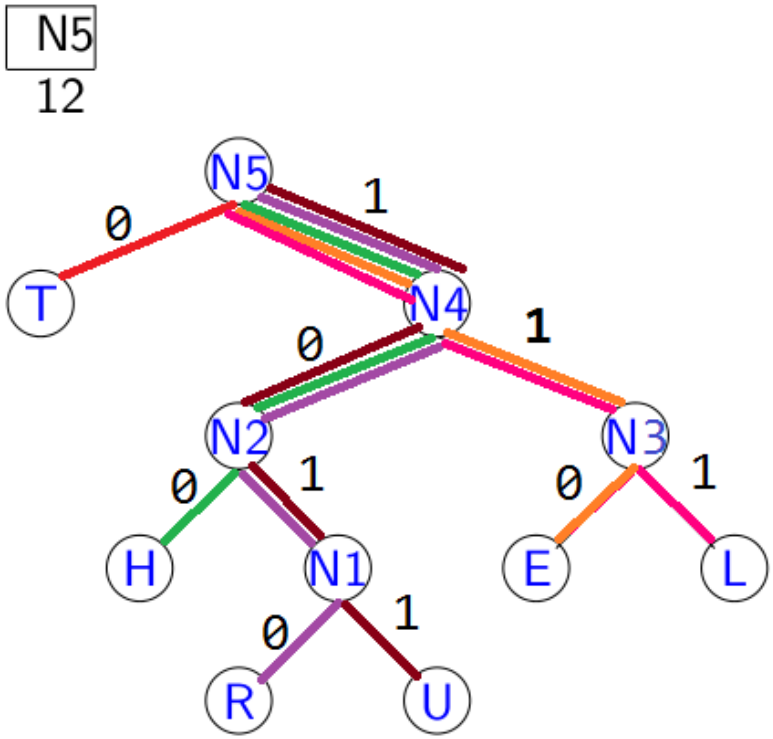
Huffman Encoding Simulation

Step: Add the last two frequency from the table and denote e new symbol.

N5
12



Huffman Encoding Simulation

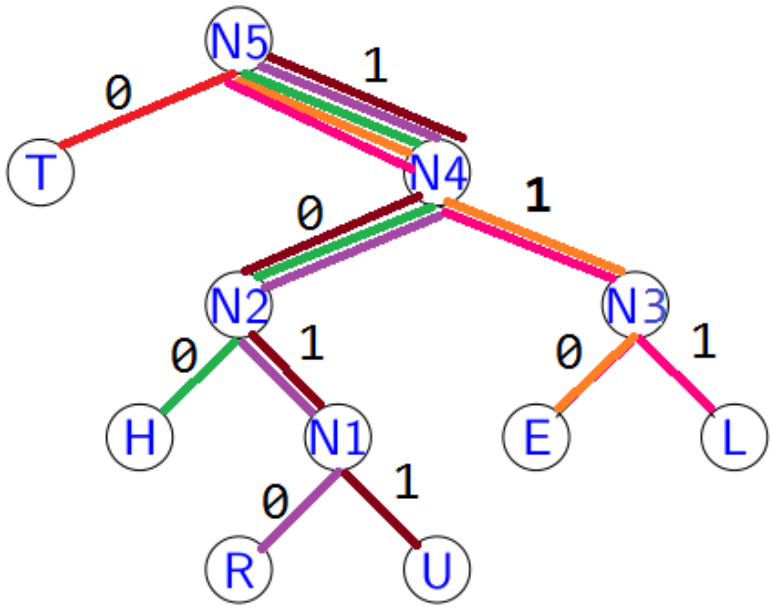


T - 0
H - 100
R - 1010
U - 1011
E - 110
L - 111

The encoded text :
01101111110100110010101010
110100

Decoding Simulation

Given code - 01101111110100110010101010110100



T - 0
H - 100
R - 1010
U - 1011
E - 110
L - 111

Decoded text -
TELL THE TRUTH

THANK YOU ALL