

Modern Education Society's  
Wadia College of Engineering, Pune

NAME OF STUDENT:	CLASS:
SEMESTER/YEAR:	ROLL NO:
DATE OF PERFORMANCE:	DATE OF SUBMISSION:
EXAMINED BY:	EXPERIMENT NO: LP-III(DAA)-02

**TITLE:** Huffman coding algorithm

**AIM:** Write a program to implement Huffman Encoding using a greedy strategy.

**OBJECTIVES:**

- 1) Analyze performance of an algorithm.
- 2) Learn how to implement algorithms that follow greedy strategy.

**PER-REQUISITES:**

1. Knowledge of programming language.
2. Knowledge of Time and space complexity

**THEORY:**

Huffman coding is a lossless data compression algorithm. The idea is to assign variable-length codes to input characters, lengths of the assigned codes are based on the frequencies of corresponding characters. The most frequent character gets the smallest code and the least frequent character gets the largest code.

**ALGORITHM:-**

***Steps to build Huffman Tree***

Input is an array of unique characters along with their frequency of occurrences and output is Huffman Tree.

1. Create a leaf node for each unique character and build a min heap of all leaf nodes (Min Heap is used as a priority queue. The value of frequency field is used to compare two nodes in min heap. Initially, the least frequent character is at root).
2. Extract two nodes with the minimum frequency from the min heap
3. Create a new internal node with a frequency equal to the sum of the two nodes frequencies. Make the first extracted node as its left child and the other extracted node as its right child. Add this node to the min heap.
4. Repeat step 2 and 3 until the heap contains only one node. The remaining node is the root node and the tree is complete.

**CONCLUSION:-**Implemented Huffman Encoding using a greedy strategy successfully.

**QUESTIONS FOR REVIEW:**

1. Explain greedy approach?
2. Explain huffman coding with example using greedy approach?
3. Analyze time complexity of huffman coding?