Greedy Algorithms

Greedy algorithm obtains the solution of a problem by making a sequence of choices. At each decision point, the best possible choice is made.

In this online, you have to implement greedy algorithm for a defined problem. You have to read the inputs from a file. Also you have to show the performance of your implementation by printing the running time.

Huffman Coding

Huffman codes are widely used and very effective techniques for data compression. The principle behind this technique is to design a binary character code for each character, using a variable number of bits per character, so as to reduce the total length of the document. For example, there be four characters a, b, c and d, and their corresponding variable length codes be 00, 01, 0 and 1.

There are mainly two major parts in Huffman Coding
1) Build a Huffman Tree from input characters.
2) Traverse the Huffman Tree and assign codes to characters.

A	C	D	E	F	G	H	I	L	N	O	R	S	T	U	V	W	X	Y	Z
3	3	2	26	5	3	8	13	2	16	9	6	27	22	2	5	8	4	5	1

Build a Huffman tree from the bottom up. The basic idea is that you have to find the two nodes with the lowest frequencies, from a tree that has each node as a child, remove the original two nodes and add the new tree.

List the costs, i.e., the required number of bits for encoding each character in the above table.

N.B.: Different Huffman codes would assign different codes, possibly with different lengths, to various characters, but the overall length of the encoded message is the same for any Huffman code.