

Inversão de Huffman

Por Leopoldo Taravilse  Argentina

Timelimit: 1

Codificação Estática de Huffman é um algoritmo de codificação usado principalmente para compressão de texto. Dado um texto de determinado tamanho feito de n caracteres diferentes, o algoritmo escolhe N códigos, um para cada diferente caractere. O texto é compactado usando esses códigos. Para escolher os códigos, o algoritmo constrói uma árvore binária com N folhas. Para $N \geq 2$ a árvore pode ser construído como se segue. 1. Para cada caractere diferente no texto construa uma árvore contendo apenas um único nodo, e atribua a ele um peso coincidente com o número de ocorrências de caracteres no texto.

2. Construa um conjunto s contendo as N árvores acima.

3. Enquanto s contiver mais de uma árvore:

(a) Escolha $t_1 \in s$ com peso mínimo e remova-o de s .

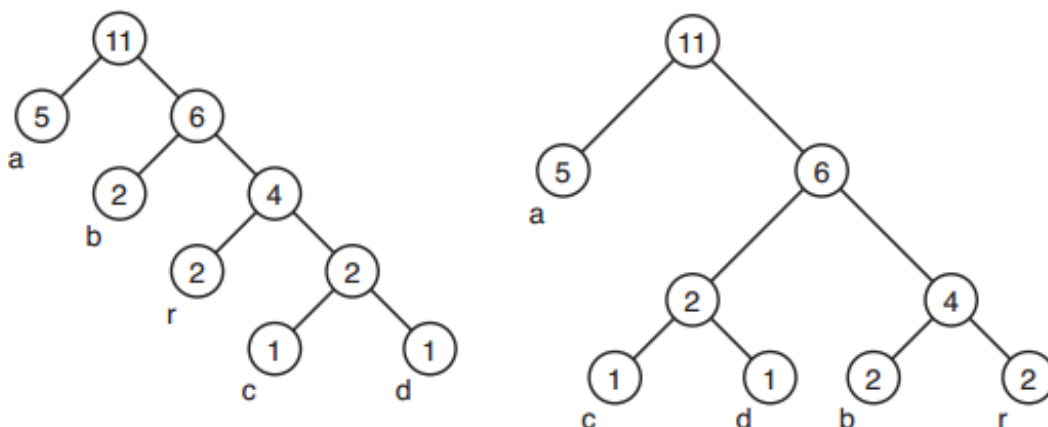
(b) Escolha $t_2 \in s$ com peso mínimo e remova-o de s .

(c) Crie uma nova árvore t com t_1 como a sua subárvore esquerda e t_2 como sua subárvore direita e atribua para t a soma dos pesos de t_1 e t_2 .

(d) Inclua t em s .

4. Retorne a única árvore que sobrar em s .

Para o texto "**abracadabra**", a árvore produzido pelo processo acima descrito pode ser semelhante a imagem à esquerda, onde cada nodo interno é marcado com o peso da sub-árvore esquerda daquele nodo. Note que a árvore obtida também pode se parecer com a imagem da direita, entre outras coisas, porque nas etapas 3a e 3b do conjunto s pode conter várias árvores com peso mínimo. Para cada caractere diferente do texto, seu código depende do caminho que existe na árvore final, à partir da raiz até a folha correspondente ao caractere. O tamanho do código é o número de arestas existentes no caminho (que coincide com o número de nodos internos do caminho). Assumindo que a árvore da esquerda foi construída por um algoritmo, o código para "**r**" tem tamanho 3 enquanto que o código para "**d**" tem tamanho 4.



Sua tarefa é, dado os tamanhos dos N códigos escolhidos pelo algoritmo, encontrar a árvore mínima (em número total de caracteres) que o texto poderá ter de forma que os códigos gerados tenham comprimento N

Entrada

A entrada contém vários casos de teste e terminam com EOF. A primeira linha de cada caso de teste contém um inteiro N ($2 \leq N \leq 50$) que representa o número de caracteres diferentes que aparece no texto. A segunda linha contém N inteiros L_i indicam os comprimentos dos códigos escolhido pelo algoritmo de Huffman para os diferentes caracteres ($1 \leq L_i \leq 50$ para $i = 1, 2, \dots, N$). você pode assumir que existe pelo menos uma

árvore, construída tal como descrito, que produz códigos com o dado comprimentos.

Saída

Para cada caso de teste de entrada, imprima uma linha com um inteiro que representa o tamanho mínimo (número total de caracteres) que o texto pode ter assim como os códigos gerados têm os comprimentos determinados.

Exemplo de Entrada	Exemplo de Saída
2 1 1 4 2 2 2 2 10 8 2 4 7 5 1 6 9 3 9	2 4 89