

# EXERCÍCIO PROGRAMA

## 1 Descrição (Ernesto Birgin)

Uma aplicação interessante de algoritmos gulosos é o Algoritmo de Huffman. Um algoritmo de determinação de códigos binários de comprimentos variados para caracteres (não é o caso que cada caractere seja representado com o mesmo número de bits), baseado na frequência de uso destes caracteres, foi sugerido por Shannon e logo depois esta ideia foi aperfeiçoada por D.A. Huffman. A ideia é associar números binários com menos bits aos caracteres mais usados num texto. Desta maneira espera-se que o número de bits economizados para codificar os caracteres que ocorrem com maior frequência em um texto seja mais que o suficiente para cobrir o déficit que ocorre ao codificarmos os caracteres que ocorrem raramente com cadeias longas de bits.

## 2 Implementação

Seu programa em C deve ler um arquivo com o texto a ser codificado. O algoritmo de Huffman que será implementado para codificar o arquivo consiste de três fases:

1. Na primeira fase a frequência de cada caractere que ocorre no texto deverá ser calculada.
2. A segunda fase do algoritmo de Huffman consiste em construir uma árvore binária baseada na frequência de uso destas letras de modo que as mais frequentemente usadas apareçam mais perto da raiz que as menos frequentemente usadas. Chamaremos esta árvore binária de árvore de Huffman. Em cada passo desta fase teremos uma coleção de árvores binárias (ou seja, uma floresta formada por árvores binárias). As folhas de cada uma destas árvores correspondem a um conjunto de caracteres que ocorrem no texto. A raiz de cada uma destas árvores será associado um número que corresponde à frequência com que os caracteres associados às folhas desta árvore ocorrem no texto. Escolhere-mos então as duas árvores desta floresta com a menor frequência e as transformaremos em uma única árvore, ligando-as a uma nova raiz cujo valor será dado pela soma dos valores das frequências das duas subárvores.
3. Finalmente na terceira fase a árvore de Huffman será usada para codificar e decodificar o texto. A árvore de Huffman construída durante a codificação será utilizada na decodificação (logo, a árvore de Huffman deverá ser armazenada junto com o texto que foi codificado).

### 3 Exemplo (CLR):

	a	b	c	d	e	f
Frequency (in thousands)	45	13	12	16	9	5
Fixed-length codeword	000	001	010	011	100	101
Variable-length codeword	0	101	100	111	1101	1100

Figure 1: Tabela

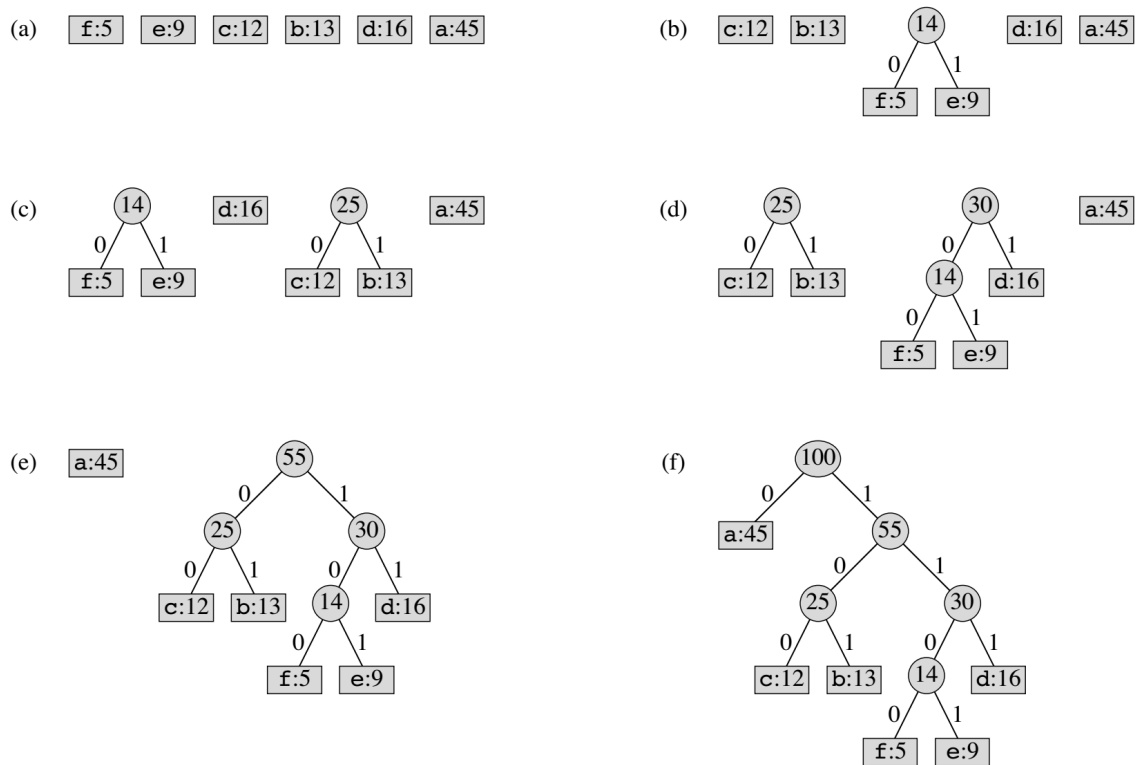


Figure 2: Exemplo

### 4 Referências e Links

1. [https://pt.wikipedia.org/wiki/Codificação\\_de\\_Huffman](https://pt.wikipedia.org/wiki/Codificação_de_Huffman)
2. Cormen, Leiserson, Rivest - Introduction to Algorithms (Chapter 16 - Greedy Algorithms - 16.3 Huffman Codes)

### 5 Observações Importantes

1. Este exercício programa pode ser feito em grupos de até 3 alunos.
2. Apenas um aluno do grupo deve submeter o programa e deve incluir o nome dos outros membros do grupo no cabeçalho do programa.