

Matemática Discreta

Adriana Padua Lovatte

Grafos e Árvores

- Grafos e Suas Representações
- Árvores e suas Representações
- Árvores de Decisão
- Códigos de Huffman

Grafos e Suas Representações

Problema: Como os arquivos são armazenados?

- 1 - O que é codificação ASCII?
- 2 - Com a codificação ASCII, quantos bits são necessário para armazenar um texto de 50.000 caracteres?
- 3 - E se esse texto for composto por apenas 6 caracteres com as seguintes frequências percentuais?

caractere	a	c	g	k	p	?
frequência	48	9	12	4	17	10

- 4 - Poderíamos gastar menos espaços utilizando para cada caractere um comprimento fixo de “n” bits por caractere?

Grafos e Suas Representações

Proposta do código de Huffman : Usar um número variável de bits e armazenar caracteres que aparecem com maior frequência como sequências com menos bits e os caracteres que aparecem menos com sequências maiores de bits.

Grafos e Suas Representações

Problema: Quanto bits são necessários usando a seguinte codificação

caractere	a	c	g	k	p	?
frequência	48	9	12	4	17	10
codificação	0	1101	101	1100	111	100

Grafos e Suas Representações

Problema: Quanto bits são necessários usando a seguinte codificação?

caractere	a	c	g	k	p	?
frequência	48	9	12	4	17	10
codificação	0	1101	101	1100	111	100

Usando a codificação acima, que cadeia de caractere geral os seguintes códigos?

- a) 11111111010100
- b) 1101010101100?
- c) 100110001101100?

Grafos e Suas Representações

caractere	a	c	g	k	p	?
frequência	48	9	12	4	17	10
codificação	0	1101	101	1100	111	100

Note que um código nunca é prefixo de outro. Isso é chamado de **código de prefixo**.

Grafos e Suas Representações

Considere a seguinte codificação:

caractere	a	b	c			
frequência						
codificação	01	101	011			

Quais são os caracteres da cadeia 01101?

OBS: Este não é código de prefixo

Grafos e Suas Representações

Armazenamento de caracteres com número variável de bits por caractere.

Prós:

Menos espaço de armazenamento;

Contras:

Necessita de conhecimento prévio do arquivo;

Na descompactação: É preciso saber uma maneira de saber quando termina a sequência de um símbolo e começa a do seguinte.

Grafos e Suas Representações

O código de prefixos estudados é construído uma árvore binária tendo caracteres como folhas;

Uma árvore de Huffman é construída recursivamente a partir da junção dos dois símbolos de menor frequência, que são então somados em símbolos auxiliares e estes símbolos auxiliares recolocados no conjunto de símbolos. O processo termina quando todos os símbolos forem unidos em símbolos auxiliares, formando uma árvore binária.

Grafos e Suas Representações

A árvore é então percorrida, atribuindo-se valores binários de 0 (aresta da esquerda) ou 1 (aresta da direita), e os códigos são gerados a partir desse percurso.

O total de bits utilizados para armazenar o texto é dado por:

$$\text{Total de bits} = S * \sum_{\text{folhas}} d(i)f(i)$$

Onde: $d(i)$ profundidade de cada folha (Número de bits do código para o caractere correspondente) ;

$f(i)$: Frequência de cada caractere ;

S: Número total de caracteres do texto;

Grafos e Suas Representações

Passos para construir a árvore binária

Passo 1- Contar a ocorrência de cada caractere e montar a tabela de frequência;

Passo 2 - Construir uma lista ordenada em ordem crescente, ou seja, o caractere que menos ocorre vem primeiro e o que mais ocorre vem ao final

Passo 3 – Montar a árvore binária ;

Passo 4 – Caminhar sobre a árvore binária para obter a tabela de códigos percorrendo da raiz até a folha;

Passo 5 – Criar o texto codificado;

Grafos e Suas Representações

Exemplo:

O texto tem 50.000 caracteres e é composto por apenas 6 caracteres (a,c,g,k, p,?) com as seguintes frequências percentuais (48,9,12,4,17,10), respectivamente.

Passo 1 :

caractere	a	c	g	k	p	?
Frequência	48	9	12	4	17	10

Grafos e Suas Representações

Passo 2 :

caractere	k	c	?	g	p	a
Frequência	4	9	10	12	17	48

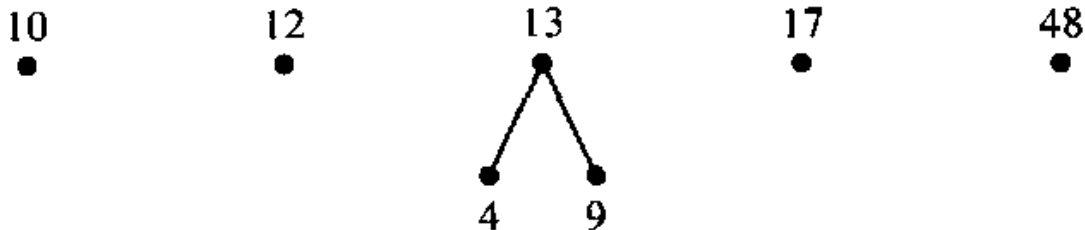
Passo 3 – Construção da árvore binária
Inicialmente a lista contém 6 caracteres

4 9 10 12 17 48

● ● ● ● ● ●

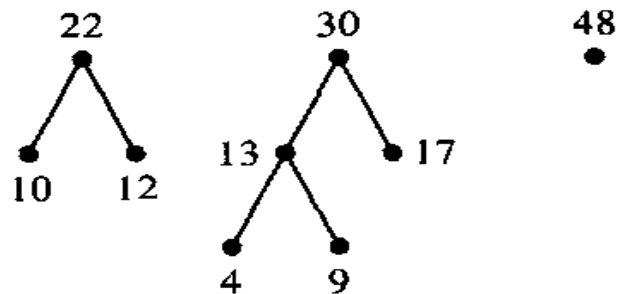
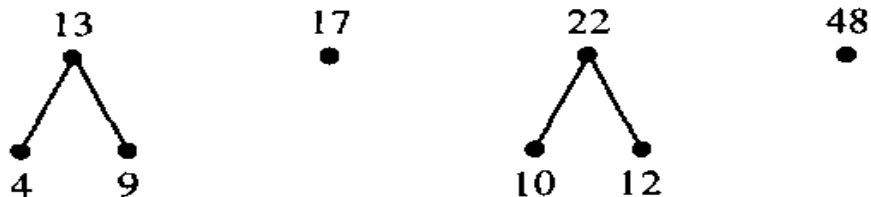
Grafos e Suas Representações

Entramos no laço **for** pela primeira vez. Os vértices x e y são os com frequência 4 e 9, respectivamente. Um novo vértice z com frequência $4 + 9 = 13$ é criado e inserido ordenadamente em L



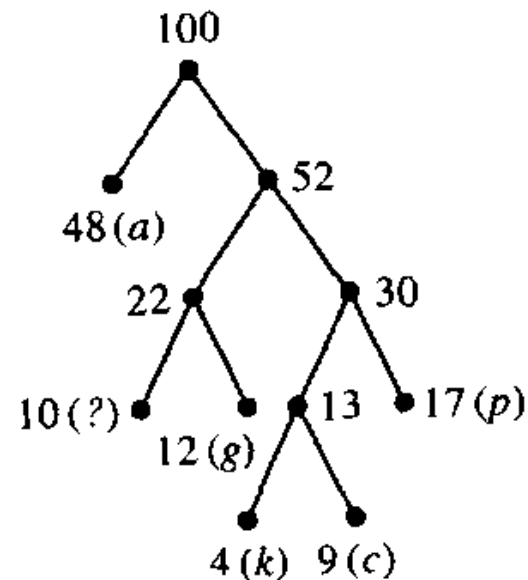
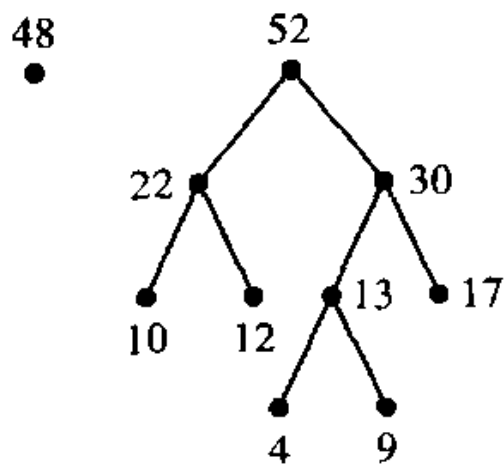
Grafos e Suas Representações

Este processo é repetido mais quatro vezes. A lista L resultante após cada passo é mostrada abaixo:



Grafos e Suas Representações

Árvore completa



Grafos e Suas Representações

- Passo 4 – Tabela de Códigos da árvore abaixo

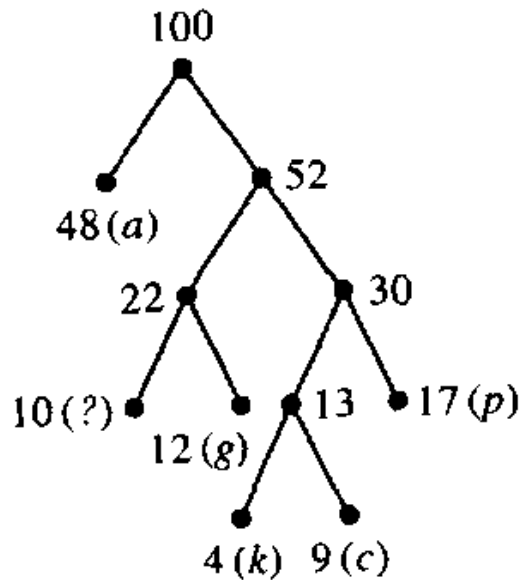


TABELA DE CÓDIGOS	
CARACTERE	CÓDIGO
a	0
?	100
g	101
k	1100
c	1101
p	111

Grafos e Suas Representações

Compactação:

Com o texto comum e a tabela de código crie o TEXTO CODIFICADO

Na descompactação:

Com o arquivo CODIFICADO e a TABELADECÓDIGO, pode-se obter o TEXTOCOMUM.

Algoritmo de Codificação de Huffman

Árvore de Huffman (Lista de nós L ; inteiro m)

// A cada um dos m nós em L está associada uma frequência f , e L é ordenada
// pela frequência em ordem crescente; algoritmo constrói a árvore de Huffman

para $i = 1$ **até** $m - 1$ **faça**

crie novo nó z

// sejam x, y os dois primeiros nós em L // nós de frequência mínima

$f(z) = f(x) + f(y)$

insira z em ordem em L

filho esquerdo de $z =$ nó x

filho direito de $z =$ nó y

fim do para

fim da Árvore de Huffman

Algoritmo de Codificação de Huffman

Processo para o uso do código de Huffman

- 1) Com o TEXTOCOMUM, faça uma análise da frequência dos caracteres usados.
- 2) Com o TEXTOCOMUM e FREQUÊNCIA, crie a TABELADECÓDIGO.
- 3) Usando TEXTOCOMUM e TABELADECÓDIGO crie o texto CODIFICADO.
- 4) Assim, com o arquivo CODIFICADO e a TABELADECÓDIGO, pode-se obter o TEXTOCOMUM.

Lista Mínima de Exercícios

Seção 5.4: 1, 3, 5, 7, 8, 10, 13, 14