Compressão de Dados

Elisa M. Pivetta Cantarelli

elisa@fw.uri.br

Compressão de Dados

- Compressão de dados é uma forma de codificar um certo conjunto de informações de maneira que o código gerado seja menor que o fonte.
- **Para que comprimir?**
  - Para redução do espaço físico utilizado
  - Para agilização na transmissão de dados.

Compressão de Dados

- A redução do espaço físico permite um significativo ganho em termos de ocupação em disco e velocidade de acesso.

Compressão

- A compressão é aquela realizada sobre dados, a partir dos quais é verificada a repetição de caracteres para efetivar a redução do número de elementos de dados.

Técnica de Compressão sem perda

- Se a informação, após sua compressão, pode ser exatamente reconstruída a técnica de compressão é dita sem perdas.
- Estas técnicas exploram estatísticas de dados (redundância de dados)
- Como exemplo de técnicas sem perda temos: codificaçãoRun-length e codificação Huffman.

8 - Compressão Estatística

- A idéia da compressão estatística é realizar uma representação otimizada de caracteres ou grupos de caracteres.
- Caracteres de maior freqüência de utilização são representados por códigos binários pequenos, e os de menor freqüência são representados por códigos proporcionalmente maiores.

8 - Compressão Estatística

- Neste tipo de compressão portanto, não necessitamos saber qual caracter vai ser comprimido, mas é necessário, porém, ter o conhecimento da probabilidade de ocorrência de todos os caracteres sujeitos à compressão.
- Caso não seja possível a tabulação de todos os caracteres sujeitos à compressão, utiliza-se uma técnica adequada para levantamento estatístico dos dados a comprimir, formando tabelas de probabilidades.

Codificação Huffman

- Esta técnica de compressão foi criada por D. A. Huffman em 1950
- No envio de uma mensagem de um dispositivo origem a um dispositivo destino os caracteres da mensagem são enviados um a um modificados através de alguma tabela de codificação.
- Em geral, este código forma um número binário.

Codificação Huffman

- Como a velocidade de transmissão é importante, é interessante tornar a mensagem tão curta quanto possível sem, é claro, perder a capacidade de decodificar o texto enviado.

- Um algoritmo de determinação de códigos binários para caracteres baseado na freqüência de uso destes caracteres foi sugerida por Huffman.

- A idéia é associar números binários com menos bits aos caracteres mais usados nos textos.

Codificação Huffman

- Neste método de compressão, é atribuído menos bits a símbolos que aparecem mais freqüentemente e mais bits para símbolos que aparecem menos.
- Codificação de Huffman é um exemplo de técnica de codificação estatística, que diz respeito ao uso de um código curto para representar símbolos comuns, e códigos longos para representar símbolos pouco freqüentes.
- Esse é o princípio do código Morse, onde:
  - E = ?
  - Q = --?-

Codificação Huffman

- Sabemos que todo caracter é formado por sequencia de 8 bits (1 byte). A idéia para a compressão é criar seqüências de bits de tamanhos variáveis, onde os caracteres que tiverem mais freqüência no arquivo usarão uma seqüência de bits menor e os caracteres que tiverem mais freqüência no arquivo usarão uma seqüência de bits maior.
- Assim para um arquivo grande ocorrerá uma diminuição no seu tamanho.

Codificação Huffman

- Uma aplicação interessante de árvores binárias em codificação de dados a serem transmitidos e compactação de arquivos são os códigos de Huffman.

Codificação Huffman

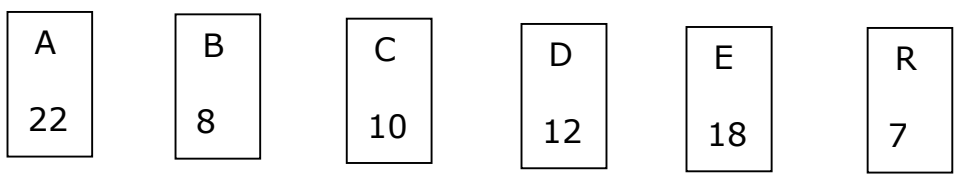
- Exemplo:
  - suponha que uma mensagem seja composta pelos caracteres A,B,C,D,E, R e que a freqüência de uso destas letras na mensagem seja 22, 8, 10, 12, 18, 7.
- A técnica de Huffman consiste em construir uma árvore binária baseada na freqüência de uso das letras de tal forma que as mais freqüentemente usadas (A, E) apareçam mais perto da raiz que as menos freqüentemente usadas (B, R).

Codificação Huffman

- A construção desta árvore binária será feita de baixo para cima, começando a partir das letras menos usadas até atingir a raiz.
- Nesta árvore binária, as letras serão representadas nas folhas e o seus vértices internos conterão um número correspondente à soma das freqüências dos seus descendentes.

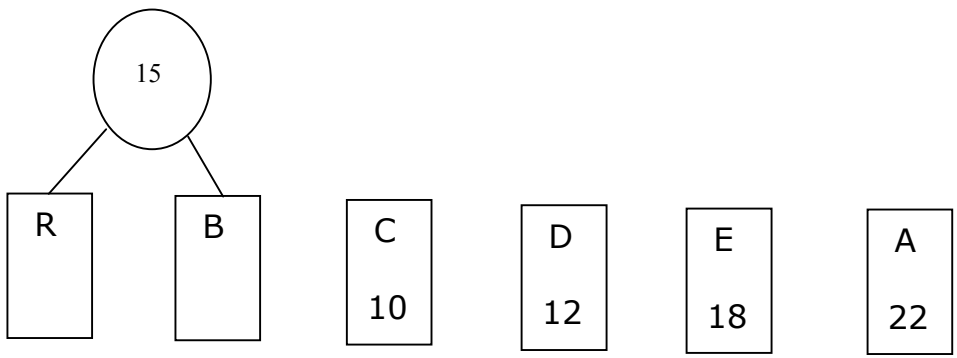
Codificação Huffman

- Em cada passo do algoritmo teremos uma coleção de árvores de Hoffman da qual tomaremos as duas com menor valor associado e as transformaremos num só, cujo valor é a soma dos valores dos descendentes.
- Inicialmente, cada uma das letras será uma árvore composta apenas pela raiz e cujo conteúdo é o valor da freqüência de uso.



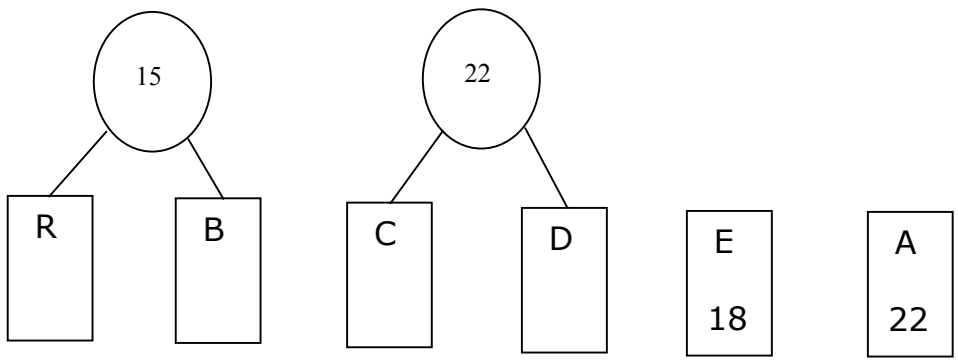
Codificação Huffman

- Selecionamos, então, as duas raízes de menor valor (7 e 8) e as juntamos em uma nova árvore, que fará parte da nossa coleção.



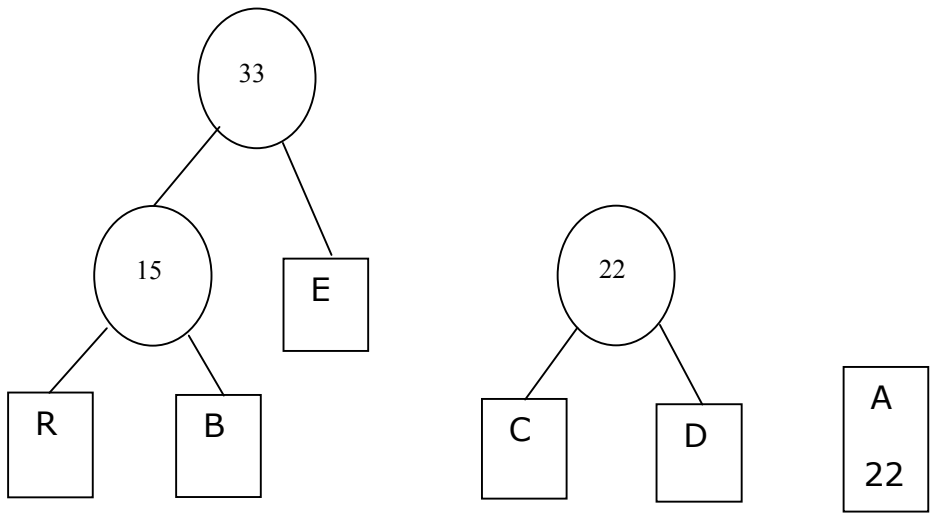
Codificação Huffman

- Repetindo o processo, escolhemos as árvores com raízes 10 e 12, obtendo uma 22 na raiz. Neste ponto, nossa coleção de árvores tem a forma:



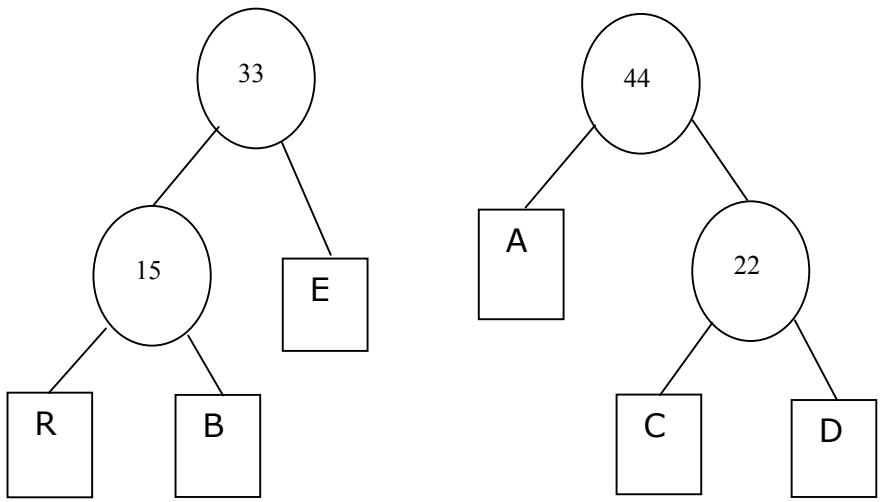
Codificação Huffman

- Continuando o mesmo processo escolhemos 18 e 15, formando:



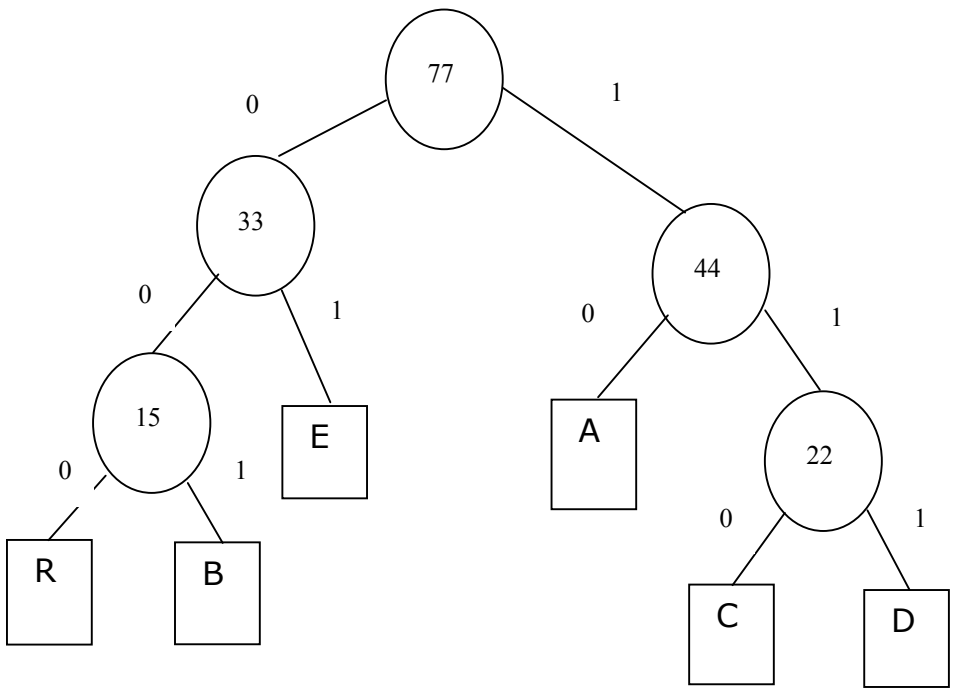
Codificação Huffman

- Depois, 22 e 22



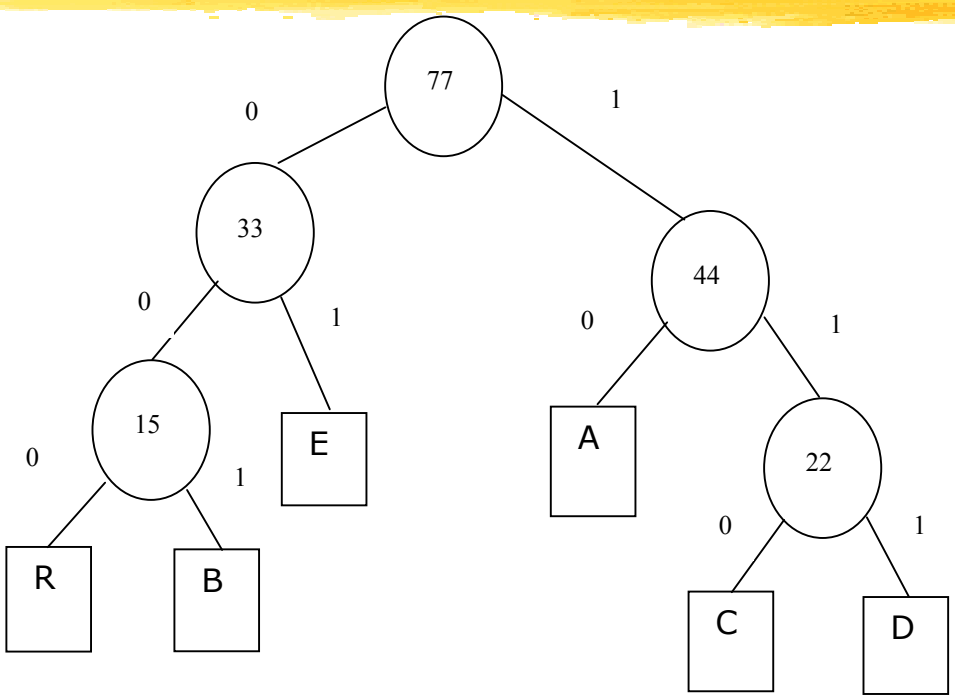
Codificação Huffman

- E, finalmente:



Temos, então, a árvore de Huffman completamente construída.
Associamos 0 às arestas que ligam um vértice com seu filho esquerdo e 1 às arestas que ligam um vértice com seu filho à direito.
O código correspondente a cada letra será formado pelo número binário associado ao caminho da raiz até a folha correspondente.

Codificação Huffman



Com isso, obtemos a tabela:

A	B	C	D	E	R
10	001	110	111	01	000

Codificação Huffman

Para decodificar uma mensagem obtida através da tabela acima por exemplo:

10 001 000 10 110 10 111 10 001 000 10

- basta ir utilizando cada bit da seqüência para percorrer a árvore de Huffman desde a raiz até se atingir uma folha, quando se obtém o caracter correspondente. Devemos, então voltar à raiz e continuar a percorrer a árvore para continuar a decodificação.
- No exemplo da string de bits acima temos a mensagem ABRACADABRA. Veja como esta mensagem foi compactada: de 11 bytes (88 bits) para 28 bits.

Codificação Huffman

- Assim acontece com os demais caracteres, em ordem crescente do número de bits, conforme a prioridade.
- A codificação Huffman manteve a propriedade de permitir a decodificação direta, não permitindo que os bits da codificação de um caracter confundisse com a de outro.