UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

DEPARTAMENTO DE ESTATÍSTICA E INFORMÁTICA

MATEMÁTICA DISCRETA – 2011.2

Prof. Marcelo Gama

3ª LISTA DE EXERCÍCIOS

1. Seja $S = \{0, 1, 2, 4, 6\}$. Verifique se as relações binárias em S são reflexivas, simétricas, anti-simétricas e/ou transitivas:

   (a) $R_1 = \{(0,0), (1,1), (2,2), (4,4), (6,6), (0,1), (1,2), (2,4), (4,6)\}$

   (b) $R_2 = \{(0,1), (1,0), (2,4), (4,2), (4,6), (6,4)\}$

   (c) $R_3 = \{(0,1), (1,2), (0,2), (2,0), (2,1), (1,0), (0,0), (1,1), (2,2)\}$

   (d) $R_4 = \{(0,0), (1,1), (2,2), (4,4), (6,6), (4,6), (6,4)\}$

   (e) $R_5 = \emptyset$

2. Classifique as relações binárias a seguir nos conjuntos S dados como reflexivas, simétricas, anti-simétricas e transitivas:

   (a) $S = \mathbb{Q}$
   $xRy \Leftrightarrow x \leq y$

   (b) $S = \mathbb{Z}$
   $xRy \Leftrightarrow x - y$ é múltiplo de 3

   (c) $S = \mathbb{Z}$
   $xRy \Leftrightarrow x \cdot y$ é par

   (d) $S = \mathbb{N}$
   $xRy \Leftrightarrow x$ é impar

   (e) $S = \mathscr{P}(\{1, 2, 3, 4, 5, 6, 7, 8, 9\})$
   $xRy \Leftrightarrow n(x) = n(y)$

   (f) $S = \mathscr{P}(\{1, 2, 3, 4, 5, 6, 7, 8, 9\})$
   $xRy \Leftrightarrow x \neq y$

   (e) $S =$ conjunto de todos os quadrados do plano
   $xRy \Leftrightarrow$ tamanho do lado de $x =$ tamanho do lado de y

   (f) $S =$ conjunto de todas as cadeias finitas de caracteres
   $xRy \Leftrightarrow$ número de caracteres em $x =$ número de caracteres em y

   (g) $S =$ conjunto de todas as pessoas do Brasil
   $xRy \Leftrightarrow x$ é irmão de y

   (j) $S = \mathbb{N} \times \mathbb{N}$
   $(x_1, y_1)R(x_2, y_2) \Leftrightarrow x_1 \leq x_2$ e $y_1 \geq y_2$

3. Quais das relações binárias do exercício anterior são relações de equivalência? Para as que o forem, descreva as classes de equivalência associadas.

4. Encontre os fechos reflexivos, simétricos e transitivos das relações do Exercício 1.

5. Para cada caso abaixo, apresente um conjunto S e uma relação binária R em S (diferente das apresentadas nos problemas anteriores) que satisfaça às condições pedidas.

   (a) R é reflexiva e anti-simétrica, mas não é transitiva.
   (b) R é reflexiva e transitiva, mas não é simétrica.
   (c) R não é reflexiva nem simétrica, mas é transitiva.
   (d) R é reflexiva, mas não é simétrica nem transitiva.

6. Desenhe o grafo das seguintes ordenações parciais:

   (a) $S = \{a, b, c\}, \qquad R = \{(a, a), (b, b), (c, c), (a, b), (a, c), (b, c)\}$
   (b) $S = \{a, b, c, d\}, \qquad R = \{(a, a), (b, b), (c, c), (d, d), (a, b), (a, c)\}$
   (c) $S = \{\emptyset, \{a\}, \{a, b\}, \{c\}, \{a, c\}, \{b\}\}, \qquad xRy \Leftrightarrow x \subseteq y$

7. Indique os elementos mínimos, minimais, máximos e maximais que aparecem nas ordenações parciais do Exercício 6.

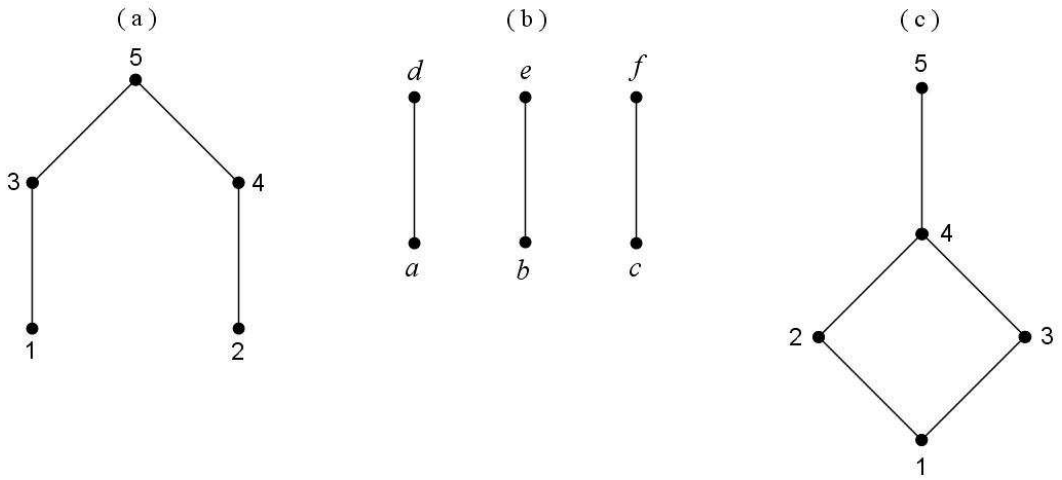
8. Desenhe o grafo dos dois conjuntos parcialmente ordenados.

   (a) $S = \{1, 2, 3, 5, 6, 10, 15, 30\}$
   $xRy \Leftrightarrow x$ divide y

   (b) $S = \mathscr{P}(\{1, 2, 3\})$
   $xRy \Leftrightarrow x \subseteq y$

   O que você pode notar a respeito da estrutura desses dois grafos?

9. Para cada grafo de ordenação parcial apresentado abaixo, escreva os pares ordenados que pertencem à relação.



10. Um programa de computador para gerar o dicionário ou o índice de um livro será escrito. Assumiremos um tamanho máximo de n caracteres por palavra. Temos, portanto, um conjunto S com palavras de, no máximo, n caracteres e desejamos gerar uma lista ordenada alfabeticamente com estas palavras. Existe a ordem natural dos caracteres do alfabeto ($a \preceq b \preceq c \preceq \ldots \preceq z$) e admitimos que nossas palavras contenham apenas caracteres alfabéticos. Desejamos definir uma ordenação total em S, chamada "ordenação lexicográfica", que ordene S alfabeticamente. A idéia é comparar duas palavras X e Y caracter a caracter, ignorando os caracteres iguais:

❶ Se, em algum momento, o caracter da palavra X precede o caracter correspondente da palavra Y, então X precede Y ($X \preceq Y$).

❷ Se todos os caracteres de X forem iguais aos caracteres correspondentes de Y, mas os caracteres de X acabaram antes dos de Y, então X precede Y ($X \preceq Y$).

❸ Caso contrário, Y precede X ($Y \preceq X$).

Formalmente: Sejam $X = x_1x_2 \ldots x_j$ e $Y = y_1y_2 \ldots y_k$, com $j \leq k$.

❶ $(x_1 \preceq y_1)$ ou $(x_1 = y_1, \ldots, x_r = y_r,\ x_{r+1} \preceq y_{r+1}) \Rightarrow X \preceq Y$.

❷ $x_1 = y_1, \ldots, x_j = y_j$ e $j < k \Rightarrow X \preceq Y$.

❸ Caso contrário, $Y \preceq X$.

(a) Mostre que $\preceq$, conforme definido acima, é uma ordenação total em S.

(b) Aplique a ordenação total descrita às palavras *roupa*, *rua*, *remédio*, *rato*, *ruga* e *ruanda*.

11. O exercício anterior abordou uma ordenação total em um conjunto de palavras com no máximo n caracteres de tamanho que gera, como saída, uma lista linear ordenada alfabeticamente. Suponha que desejamos gerar uma lista com todas as palavras distintas do texto na ordem em que aparecem no mesmo (por exemplo, um compilador precisa gerar uma tabela de símbolos com os nomes das variáveis). Como no exercício anterior, assumimos que as palavras contêm apenas caracteres alfabéticos porque já existe uma relação natural de precedência ($a < b < c < \ldots < y < z$). Caso sejam permitidos caracteres numéricos ou especiais, eles precisam ter uma relação de precedência junto aos caracteres alfabéticos (a sequência de ordenação precisa ser definida). Vamos analisar duas maneiras de gerar essa lista:

❶ Listando as palavras em ordem alfabética, o procedimento para decidir se uma palavra sendo processada é nova é bem simples, mas para colocar a nova palavra no lugar certo precisamos mover todas as palavras após ela uma linha para baixo.

❷ Listando as palavras na ordem em que forem processadas, as novas palavras podem ser simplesmente incluídas ao fim da lista sem a necessidade de qualquer rearrumação mas, para determinar se a palavra sendo processada é nova ou não, é preciso compará-la com todas as outras palavras da lista.

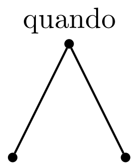
Portanto, ambos os tipos de lista apresentam suas desvantagens.

Descreveremos um processo de enumeração que se vale de uma **árvore binária de busca** que permite, para o caso geral, determinar de forma rápida se uma palavra é nova e, se este for o caso, não há a necessidade de realizar uma rearrumação para alocá-la em seu lugar. Dessa forma, combinanamos as vantagens dos dois métodos descritos acima e eliminamos suas desvantagens.

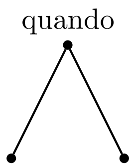
Suponha que desejamos processar a frase “Quando vimos já não era mais possível”. A primeira palavra da frase é usada para dar nome ao primeiro vértice de um grafo.

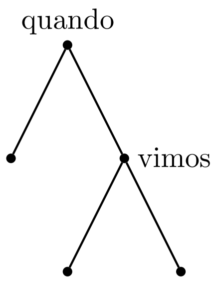
• quando

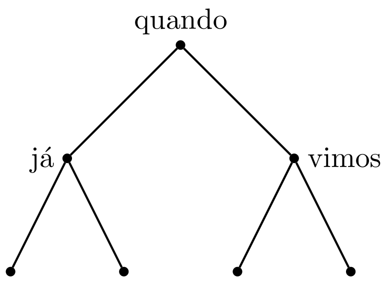
Uma vez que um vértice tenha recebido seu nome, ele recebe duas arestas para baixo, que levam a dois vértices sem nome.

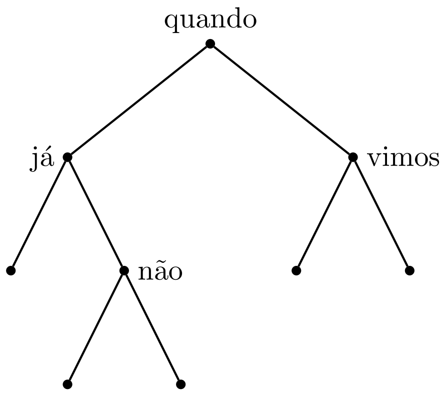


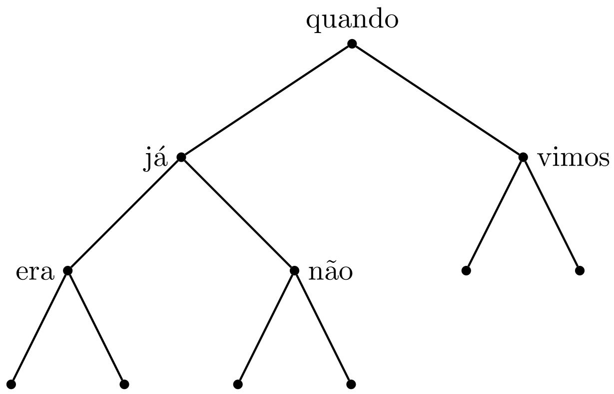
Quando a próxima palavra no texto for processada, ela é comparada com o primeiro nó. Se a palavra sendo processada preceder a palavra que dá nome ao vértice, tomamos a aresta da esquerda, do contrário tomamos a aresta da direita. A palavra que em questão torna-se o nome do primeiro vértice ainda sem nome que for encontrado. Caso a palavra seja igual ao nome de algum vértice, significa que aquela palavra já havia aparecido no texto e passamos ao processamento da palavra seguinte. Este procedimento é repetido para todo o texto. Desta forma, obtemos

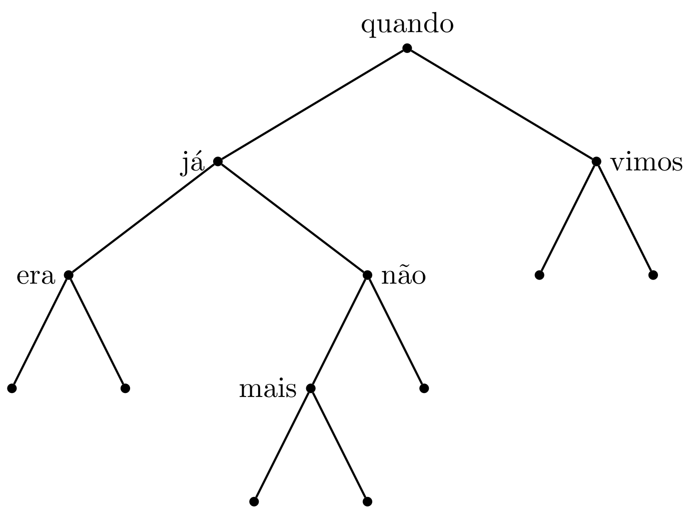


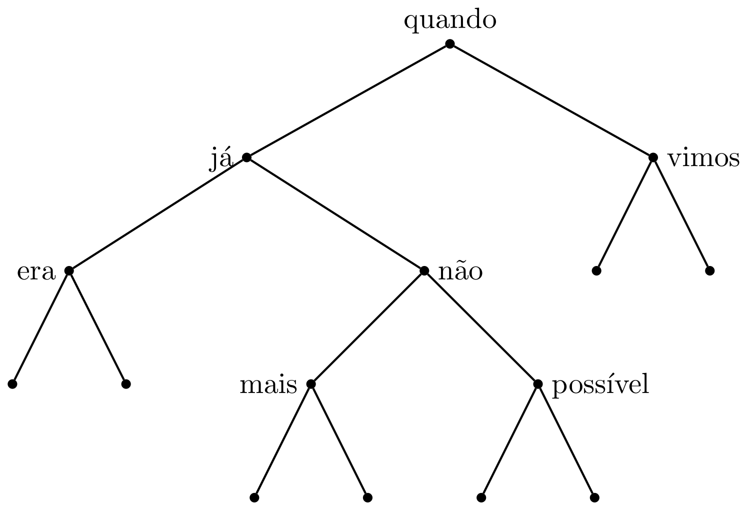












Varrerndo os vértices deste grafo na ordem apropriada (definida sempre como processar os vértices à esquerda antes, depois o próprio vértice e, depois os vértices à direita) obtemos uma lista em ordem alfabética "era, já, mais, não, possível, quando, vimos".

(a) Este tipo de grafo é chamado de árvore. Virando-o de cabeça para baixo, podemos entendê-lo como uma ordenação parcial $\preceq$. Qual seria o elemento mínino? Há um elemento máximo?

(b) Quais dos pares a seguir pertenceriam a $\preceq$? (não, vimos), (já, era), (já, mais), (era, possível)

Observação:
Neste caso, a estrutura de árvore contém mais informação do que a ordenação parcial pois nos diz não só quando uma palavra w_1 precede uma palavra w_2, mas também que w_2 está à esquerda ou à direita de w_1.

(c) Use uma árvore binária de busca para desenhar o grafo de "Suco de laranja faz bem à saúde". Em seguida, considerando o grafo (de cabeça para baixo) como uma ordenação parcial, determine os elementos maximais.

12. Responda às seguintes questões:

(a) Qual o conjunto $[a]$ para a relação de equivalência $R_1 = \{(a,a),(b,b),(c,c),(a,c),(c,a)\}$? Ele tem outras representações?

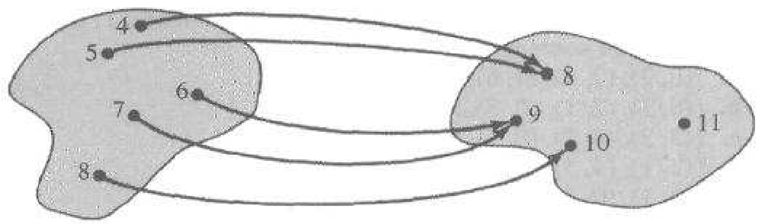
(b) Quais os conjuntos [3] e [4] para a relação de equivalência

$$R_2 = \{(1,1),(2,2),(1,2),(2,1),(1,3),(3,1),(3,2),(2,3),(3,3),(4,4),(5,5),(4,5),(5,4)\}$$

(c) Dada a partição $\{1,2\}$ e $\{3,4\}$ do conjunto $S = \{1,2,3,4\}$, liste os pares ordenados da relação de equivalência correspondente.

(d) Dada a partição $\{a,b,c\}$ e $\{d,e\}$ do conjunto $S = \{a,b,c,d,e\}$, liste os pares ordenados da relação de equivalência correspondente.

13. A figura a seguir representa uma função f.



(a) Qual seu domínio? Qual seu contradomínio? Qual o conjunto imagem?

(b) Qual a imagem de 5? E de 8?

(c) Quais as pré-imagens de 9?

(d) Esta função é sobrejetiva? É injetiva?

14. Uma "pilha" é uma estrutura de armazenamento de dados cuja operação é muito parecida com uma pilha de pratos na pia de um restaurante. Todos os locais de armazenamento começam vazios. Um elemento de dado é incluído no topo da pilha através da instrução push, que "empurra" todos os itens já empilhados uma posição para baixo a fim de abrir espaço para ele. Apenas o elemento no topo da pilha pode ser acessado a qualquer momento e ele pode ser examinado e removido da pilha através da instrução pop.

    Consideremos cadeias de inteiros onde os caracteres são inteiros positivos ou zeros. Processaremos essas cadeias através de uma função de armazenamento na forma de pilha do seguinte modo:

    > A função toma uma cadeia de inteiros não negativos. À medida que lermos da esquerda para direita, aplicamos a instrução push a qualquer caracter diferente de zero. Caracteres zero causam a instrução pop na pilha. A saída produzida pela função é a sequência de caracteres que restaram na pilha (do topo para baixo) ou zero se a pilha ficar fazia.

    Portanto, o processamento da cadeia 120304 resulta na saída 41, enquanto que o processamento da cadeia 12304000 resulta na saída 0. (Uma cadeia como 10020340 não pode ser tratada por este procedimento porque não podemos efetuar a instrução pop quando a pilha estiver vazia.)

    (a) Qual o domínio dessa função?

    (b) Qual o conjunto imagem?

    (c) Essa função é injetiva? É sobrejetiva?

    (d) Encontre uma cadeia que não possa ser processada mas que possa ser processada se escrita "de trás para frente".