Universidade Federal da Paraíba

CCAE - Depto. de Ciências Exatas (DCX)

Matemática Elementar - **Resumo das Aulas**

UNIDADE III

Prof. Laudelino (laudelino@dcx.ufpb.br)

Sumário

1 Recorrência

Uma definição por recorrência é formada por duas etapas:

1. Uma base, ou condição básica, onde alguns casos são descritos explicitamente (ponto de partida).

2. Um passo de recorrência, onde novos casos são definidos em função de casos anteriores.

1.1 Sequências definidas por recorrência

Uma sequência S é um conjunto de objetos numerados em uma determinada ordem, ou seja, a ordem em que se escreve os elementos do conjunto S tem importância.

Exemplo 1.1 $S(1) = 2$ *(base)*, $S(n) = 2S(n-1), n \geq 2$ *(passo de recorrência)*

$$\begin{array}{l|l} S(2) = 2S(2-1) = 2S(1) = 2 \cdot 2 = 4 & S(3) = 2S(3-1) = 2S(2) = 2 \cdot 4 = 8 \\ S(4) = 2S(4-1) = 2S(3) = 2 \cdot 8 = 16 & S(5) = 2S(5-1) = 2S(4) = 2 \cdot 16 = 32 \end{array}$$

$$S : 2, 4, 8, 16, 32, 64, 128, \ldots$$

Exemplo 1.2 $T(1) = 1$ *(base)*, $T(n) = T(n-1) + 3, n \geq 2$ *(passo de recorrência)*

$T(2) = T(1) + 3 = 1 + 3 = 4$, $T(3) = T(2) + 3 = 4 + 3 = 7$

$$T : 1, 4, 7, 10, 13, 16, 19, 21, \ldots$$

Exemplo 1.3 *Sequência de Fibonacci:* $F(1) = 1$ *e* $F(2) = 1$ *(base)*, $F(n) = F(n-2) + F(n-1), n \geq 3$ *(passo de recorrência)*

$F(3) = F(1) + F(2) = 1 + 1 = 2$, $F(4) = F(2) + F(3) = 1 + 2 = 3$

$$F : \underbrace{1, 1}_{\text{base}}, 2, 3, 5, 8, 13, 21, 34, \ldots$$

1.2 Conjuntos definidos por recorrência

No caso de sequências, existe uma ordem para listar os elementos, no caso de conjuntos, como já estudamos, não precisamos seguir uma ordem quando listamos seus elementos.

Exemplo 1.4 *Seja A um conjunto tal que $2, 3 \in A$ (base) e $x, y \in A \Rightarrow x^y \in A$ (passo de recorrência). Descreva alguns elementos do conjunto A.*

$2, 3 \in A \Rightarrow 2^3 = 8 \in A$	$3, 2 \in A \Rightarrow 3^2 = 9 \in A$	$2, 8 \in A \Rightarrow 2^8 = 256 \in A$
$8, 2 \in A \Rightarrow 8^2 = 64 \in A$	$8, 3 \in A \Rightarrow 8^3 = 512 \in A$	$9, 2 \in A \Rightarrow 9^2 = 81 \in A$

$$A = \{2, 3, 8, 9, 256, 64, 512, 81, \ldots\}$$

Exemplo 1.5 *Seja B um conjunto tal que $4, 5 \in B$ (base) e $x, y \in B \Rightarrow x + y + 2 \in B$ (passo de recorrência).*

$4, 5 \in B \Rightarrow 4 + 5 + 2 = 11 \in B$	$4, 11 \in B \Rightarrow 4 + 11 + 2 = 17 \in B$
$5, 11 \in B \Rightarrow 5 + 11 + 2 = 18 \in B$	$11, 18 \in B \Rightarrow 11 + 18 + 2 = 31 \in B$
$11, 17 \in B \Rightarrow 11 + 17 + 2 = 30 \in B$	$11, 31 \in B \Rightarrow 11 + 31 + 2 = 44 \in B$

$$B = \{4, 5, 11, 17, 18, 31, 30, 44, \ldots\}$$

1.3 Exercícios (Recorrência)

1.1 *Determine os cinco primeiros termos das sequências:*

(a) $Q(1) = 2, \quad Q(n) = 2Q(n-1), \; n \geq 2;$

(b) $T(1) = 1, \quad T(n) = 3 + T(n-1), \; n \geq 2;$

(c) $A(1) = 2, \quad A(n) = \frac{1}{A(n-1)}, \; n \geq 2;$

(d) $M(1) = 2, \quad M(2) = 2, \quad M(n) = 2M(n-1) + M(n-2), n \geq 3;$

(e) $D(1) = 3, \quad D(2) = 5, \quad D(n) = (n-1)D(n-1) + (n-2)D(n-2), \; n \geq 3;$

(f) $W(1) = 2, \quad W(2) = 3, \quad W(n) = W(n-1)W(n-2), \; n \geq 3;$

(g) $S(1) = 9, \; S(2) = 6$ *e* $S(n) = \dfrac{1}{2}S(n-1) - \dfrac{2}{3}S(n-2), \; n \geq 3;$

(h) $R(1) = 1, \quad R(2) = 2, \quad R(3) = 3, \quad R(n) = R(n-1) + 2R(n-2) + 3R(n-3), \; n \geq 4.$

2 Indução

Seja $\mathbb{N} = \{1, 2, 3, 4, ...\}$ = conjunto dos números naturais e P uma propriedade. Queremos verificar se a propriedade P é válida para todos os números naturais. Para tanto, devemos verificar se $P(1)$ é verdade, $P(2)$, $P(3)$ e assim por diante. Tendo em vista que $\mathbb{N}$ é infinito, não iriamos parar de fazer estas verificações. Para evitar este contratempo, utilizamos indução.

Primeiro Princípio de Indução Matemática (ou Indução Fraca)

1. Verificar se $P(1)$ é verdade. (base da indução)
2. $\underbrace{P(n) \text{ é verdade para } n > 1}_{\text{Hipótese indutiva}} \Rightarrow P(n+1)$ é verdade. (passo indutivo)

Se os itens 1 e 2 forem satisfeitos, então a propriedade P é válida para todo $n \in \mathbb{N}$.

Segundo Princípio de Indução Matemática (ou Indução Forte)

1. Verificar se $P(1)$ é verdade. (base da indução)
2. $\underbrace{P(r) \text{ é verdade para todo } 1 \leq r \leq n}_{\text{Hipótese indutiva}} \Rightarrow P(n+1)$ é verdade. (passo indutivo)

Se os itens 1 e 2 forem satisfeitos, então a propriedade P é válida para todo $n \in \mathbb{N}$.

Teorema 2.1 *Seja A um conjunto. Se A tem n elementos, então $\mathcal{P}(A)$ tem 2^n elementos.*

Demonstração: Iremos utilizar o Primeiro Princípio de Indução em n.

Neste caso, a propriedade P é

$$P(n) : |A| = n \Rightarrow |\mathcal{P}(A)| = 2^n.$$

1. Para $n = 1$, seja $A = \{e\}$, então $|A| = 1$ e $\mathcal{P}(A) = \{\varnothing, A\} \Rightarrow |\mathcal{P}(A)| = 2$. Portanto, para $n = 1$ a propriedade P é verdade.

2. Suponhamos que a propriedade P seja válida para todo conjunto A com n elementos. Sejam B um conjunto com $n+1$ elementos e $e \in B$. Então, $B - \{e\}$ é um conjunto com n elementos e, pela hipótese indutiva, $|\mathcal{P}(B - \{e\})| = 2^n$. Observamos que

$$\mathcal{P}(B - \{e\}) \subseteq \mathcal{P}(B) \text{ e } \mathcal{P}(B) = \mathcal{P}(B - \{e\}) \cup \{A \cup \{e\}; A \in \mathcal{P}(B - \{e\})\}.$$

Logo, como $\mathcal{P}(B - \{e\}) \cap \{A \cup \{e\}; A \in \mathcal{P}(B - \{e\})\} = \varnothing$, temos

$$|\mathcal{P}(B)| = |\mathcal{P}(B - \{e\})| + |\{A \cup \{e\}; A \in \mathcal{P}(B - \{e\})\}| = 2^n + 2^n = 2 \cdot 2^n = 2^{n+1}.$$

Portanto, $|B| = n + 1 \Rightarrow |\mathcal{P}(B)| = 2^{n+1}$.

Dos itens 1 e 2, concluimos que a propriedade P é válida para todo $n \in \mathbb{N}$. ■

Exemplo 2.2 *Verifique que* $1 + 2 + 3 + 4 + \ldots + n = \dfrac{n(n+1)}{2}, \forall n \in \mathbb{N}$.

Solução: 1. Para $n = 1$, temos

$$1 = \frac{1(1+1)}{2} = \frac{1 \cdot 2}{2} = 1.$$

Logo, para $n = 1$ é verdade.

2. Suponhamos verdade para $n > 1, 1 + 2 + 3 + \ldots + n = \dfrac{n(n+1)}{2}$. Vamos verificar, utilizando a hipótese indutiva, se para $n + 1$ é verdade.

$$1+2+3+\ldots+n+(n+1) = \frac{n(n+1)}{2} + (n+1) = \frac{n(n+1)}{2} + \frac{2(n+1)}{2} = \frac{(n+1)[(n+1)+1]}{2}.$$

Logo, o passo indutivo é verdade.

Dos itens 1 e 2, temos, pelo primeiro princípio da indução, que

$$1 + 2 + 3 + \ldots + n = \frac{n(n+1)}{2}, \forall n \in \mathbb{N}$$

é verdade.

Exemplo 2.3 *Verifique que* $3^{2n} + 7$ *é divisível por* 8 *para todo* $n \in \mathbb{N}$.

Solução: 1. Para $n = 1$, temos $3^{2 \cdot 1} + 7 = 3^2 + 7 = 9 + 7 = 16$ e 16 é divisível por 8. Logo, para $n = 1$ é verdade.

2. Suponhamos verdade para $n > 1$, ou seja, $3^{2n} + 7$ é divisível por 8 para algum n maior do que 1. Vamos verificar se para $n + 1$ é verdade.

$$3^{2 \cdot (n+1)} + 7 = 3^{2n+2} + 7 = 3^{2n} \cdot 3^2 + 7 = 3^{2n} \cdot 3^2 + (7 \cdot 3^2 - 7 \cdot 3^2) + 7 = 9(3^{2n} + 7) - 7 \cdot 8.$$

Como, por hipótese indutiva, $3^{2n} + 7$ é divisível por 8, então $3^{2(n+1)} + 7 = 9(3^{2n} + 7) - 7 \cdot 8$ é divisível por 8.

Portanto, dos itens 1 e 2, temos, pelo primeiro princípio da indução, que $3^{2n} + 7$ é divisível por 8 para todo $n \in \mathbb{N}$.

Exemplo 2.4 *$n^2 > n+1$ para $n \geq 2$.*

Exemplo 2.5 *$n! > n^2$ para $n \geq 4$.*

Solução: Observamos que, para n igual a $1, 2$ e 3 a propriedade não é válida (verifique). A base da indução será verificada para $n = 4$. Assim, a propriedade será válida para $n \geq 4$.

1. $n = 4$, $4! = 1 \cdot 2 \cdot 3 \cdot 4 = 24 > 16 = 4^2$. Logo, para $n = 4$ a propriedade é verdadeira.

2. Suponhamos verdade para um certo $n > 4$. Verificaremos se para $n+1$ é válido:

$$(n+1)! = n!(n+1) > n^2(n+1) = \underbrace{n^3 + n^2 > (n^2+n) + (n+1)}_{\text{Utilizamos } n^2 > n+1, \forall n \geq 2} = n^2 + 2n + 1 = (n+1)^2.$$

Portanto, de 1 e 2, temos que, pelo primeiro princípio de indução, $n! > n^2, \forall n \geq 4$.

2.1 Exercícios (Indução)

2.1 *Resolva o Exemplo 2.4.*

2.2 *Prove que $2^n > n^2,\ \forall n \geq 5,\ n \in \mathbb{N}$.*

2.3 *Prove que $2^n + (-1)^{n+1}$ é divisível por 3, para todo $n \geq 1,\ n \in \mathbb{N}$.*

2.4 *Utilizando indução, prove os itens abaixo:*

(a) $1 + 3 + 5 + \ldots + (2n-1) = n^2,\ n \geq 1;$

(b) $1^3 + 2^3 + \ldots + n^3 = (1 + 2 + \ldots + n)^2,\ n \geq 1;$

(c) $1 \cdot 2 + 2 \cdot 3 + \ldots + n(n+1) = \dfrac{n(n+1)(n+2)}{3},\ n \geq 1.$

3 Grafos

Um *grafo* G é um par $(V(G), E(G))$, onde $V(G)$ é um conjunto finito e $E(G)$ é um conjunto composto de pares não-ordenados de elementos de $V(G)$. Diremos que $V(G)$ e $E(G)$ são, respectivamente, o conjunto de *vértices* e *arestas* do grafo G.

Dois vértices v e w de G são *adjacentes* se o par formado por v e w é uma aresta de G, esta aresta é dita *incidente* sobre v e também sobre w; v e w são os *extremos* da aresta e, neste caso, denotamos e por vw. O *grau* $d_G(v)$ de um vértice v de G é a quantidade de arestas de G incidentes a este vértice.

Chamamos de *grafo simples*, um grafo que não possui *arestas em paralelo* nem *laços*.

Entendemos por *subgrafo* de um grafo G, como sendo um grafo $H = (V(H), E(H))$ tal que $V(H) \subseteq V(G)$ e $E(H) \subseteq E(G)$.

3.1 Passeios, trilhas, caminhos e circuitos

Um *passeio* em um grafo G é uma sequência finita, não-nula, $P = v_0e_1v_1...e_kv_k$ cujos termos são vértices e arestas postos de forma alternada e tais que, para $1 \leq i \leq k$, os extremos de e_i são os vértices v_{i-1} e v_i. Chamamos v_0 de *vértice inicial* e v_k de *vértice terminal* do passeio P, os vértices $v_1, v_2, ..., v_{k-1}$ são conhecidos por *vértices interiores* do passeio P. O inteiro k é o *comprimento* do passeio P. Em um grafo simples, um passeio $v_0e_1v_1...e_kv_k$ é determinado apenas pela sequência de vértices $v_0v_1...v_k$.

Se as arestas $e_1, e_2, ..., e_k$ de um passeio P são distintas, então chamamos P de *trilha* e, neste caso, o comprimento de P é $|E(P)|$. Se, além das arestas, os vértices $v_0, v_1, ..., v_k$ também são distintos, dizemos que P é um *caminho*. Chamamos um caminho de vértice inicial v_0 e vértice terminal v_k de v_0v_k-*caminho*. Um caminho em G é *Hamiltoniano* quando contém todos os vértices de G.

Um passeio é *fechado* se tem comprimento positivo e o vértice inicial é também o vértice terminal. Um caminho fechado é chamado de *circuito* (ou *ciclo*).

Exemplo 3.1 *Consideremos o grafo G_0 com o seguinte conjunto de vértices e arestas:* $V(G_0) = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$, $E(G_0) = \{v_1v_2, v_1v_3, v_2v_3, v_2v_7, v_3v_7, v_3v_4, v_4v_5, v_4v_6, v_4v_7, v_5v_6\}$.

O grafo G_0 está desenhado na Figura 1. G_0 é um grafo simples por não ter arestas em paralelo nem laços.

$v_5v_4v_3v_2$ é um passeio no grafo G_0 de comprimento 3.

$v_1v_3v_7$ é um passeio de comprimento 2.

$v_1v_2v_7v_4v_3v_1$ é um circuito de comprimento 5.

O vértice v_1 tem grau 2, escrevemos $d_{G_0}(v_1) = 2$. O vértice v_3 tem grau 4, escrevemos $d_{G_0}(v_3) = 4$. Grau dos demais vértices: $d_{G_0}(v_2) = 3, d_{G_0}(v_4) = 4, d_{G_0}(v_5) = 2, d_{G_0}(v_6) = 2, d_{G_0}(v_7) = 3$.

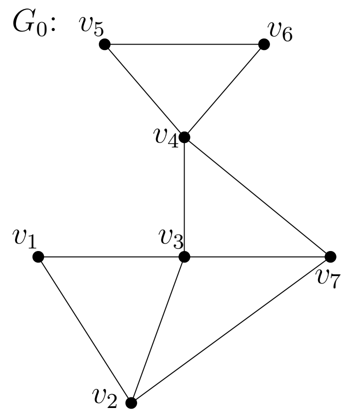


Figura 1: Grafo G_0

3.2 Alguns tipos de grafos

Grafo completo: grafo simples em que todos os vértices estão ligados entre si por uma aresta. Denotamos o grafo completo de n vértices por K_n.

Grafo conexo: um grafo G é conexo quando dados quaisquer dois vértices v e w de G, existe um vw-caminho. Em outras palavras, um grafo G é conexo quando é formado por um único pedaço.

Grafo direcionado: é um grafo tal que cada aresta possui um sentido.

Grafo com pesos (ponderado): é um grafo onde cada aresta possui um valor numérico.

3.3 Matriz de Adjacência de um grafo

Podemos representar um grafo através de uma matriz, a qual é chamada de *matriz de adjacência*.

Se um grafo G possui n vértices, então sua matriz de adjacência, denotada por $A(G)$, é da forma $n \times n$, ou seja, $A(G)$ possui n linhas e n colunas. As entradas da matriz $A(G) = (a_{ij})$

são dadas por

$$a_{ij} := \text{ número de arestas entre o vértice } v_i \text{ e o vértice } v_j.$$

Exemplo 3.2 *A matriz de adjacência do grafo G_0 exibido na Figura 1 é*

$$A(G_0) = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

3.4 Árvores

Um grafo conexo G é uma árvore quando G não possui circuitos.

O *comprimento* (ou *profundidade*) de uma árvore é o comprimento do maior caminho existente nesta árvore.

Uma *floresta* é um conjunto de árvores.

O Teorema 3.3 nos ensina maneiras de identicar uma árvore.

Teorema 3.3 *Seja T um grafo com n vértices. Então, as seguintes afirmações são equivalentes:*

1. *T é uma árvore.*
2. *T não contém circuitos e possui $n-1$ arestas.*
3. *T é conexo e possui $n-1$ arestas.*
4. *Quaisquer dois vértices de T estão conectados por exatamente um caminho.*
5. *T não possui circuitos, mas a adição de qualquer nova aresta cria exatamente um circuito.*
6. *T é conexo e a remoção de qualquer aresta torna o grafo desconexo.*

Uma *árvore geradora* de um grafo G é um subgrafo T de G tal que T é uma árvore e os vértices de T são todos os vértices do grafo G, $V(T) = V(G)$. Na seção 3.5 veremos três algoritmos para determinar árvores geradoras em um grafo.

Exemplo 3.4 *Consideremos o grafo G_0 exibido na Figura 1. Seja o grafo H_0 dado pelo conjunto de vértices e arestas:* $V(H_0) = V(G_0)$, $E(H_0) = \{v_1v_2, v_2v_3, v_2v_7, v_3v_4, v_4v_5, v_4v_6\}$. *$H_0$ é um subgrafo de G_0 e, em particular, é uma árvore geradora de G_0.*

3.5 Algoritmos para Grafos

Os algoritmos aqui descritos servem para determinar uma árvore geradora de um grafo.

3.5.1 Árvores geradoras com custo mínimo

Para um grafo com pesos (grafo ponderado) G, o custo de um subgrafo H do grafo G é definido como

$$c(H) = \sum_{e \in E(H)} c(e),$$

onde $c(e)$ é o custo da aresta e.

A seguir, será apresentado uma simplificação do algoritmo de Prim para encontrar uma árvore geradora com custo mínimo em um grafo ponderado conexo.

Algoritmo Guloso (1a. versão):

Entrada: Um grafo ponderado, conexo e simples G.

Saída: Uma árvore geradora T de G com custo mínimo.

1. Escolha vértice v de G e faça $V := \{v\}, E = \varnothing$.

2. Se $V = V(G)$, então pare e retorne T, onde $V(T) = V$ e $E(T) = E$.

3. Se $V \neq V(G)$, escolha aresta e de G incidente a um vértice v de V e a um outro w em $V(G) - V$ possuindo peso mínimo. Faça $V := V \cup \{w\}$ e $E := E \cup \{e = vw\}$. Volte à etapa 2.

Diferente do algortimo guloso, a busca em profundidade e a busca em largura são feitas para um grafo conexo simples.

3.5.2 Busca em Profundidade

Entrada: Grafo conexo e simples G.

Saída: Árvore geradora T do grafo G.

1. Escolha vértice v de G e faça $V := \{v\}$ e $E := \varnothing$.

2. Se $V = V(G)$, então pare e retorne T, onde $V(T) = V$ e $E(T) = E$.

3. Se $V \neq V(G)$ e tiver vértice $w \in V(G) - V$ adjacente a v, faça $V := V \cup \{w\}$, $E := E \cup \{vw\}$ e $v := w$. Volte à etapa 2.

4. Se $V \neq V(G)$ e não tiver vértice $w \in V(G) - V$ adjacente a v, escolha vértice $x \in V - \{v\}$ adjacente a v tal que $xv \in E$ e faça $v := x$. Volte à etapa 3.

3.5.3 Busca em Largura (ou Nível)

Entrada: Grafo conexo e simples G.

Saída: Árvore geradora T do grafo G.

1. Ordene os vértices de G, escolha vértice v de G e faça $V := \{v\}$ e $E := \varnothing$.

2. Se $V = V(G)$, então pare e retorne T, onde $V(T) = V$ e $E(T) = E$.

3. Se $V \neq V(G)$ e tiver algum vértice pertencente a $V(G) - V$ adjacente a v, escolha todos os vértices $v_1, v_2, \ldots, v_n \in V(G) - V$ adjacentes a v e faça $V := V \cup \{v_1, v_2, \ldots, v_n\}$, $E := E \cup \{vv_1, vv_2, \ldots, vv_n\}$, escolha o menor vértice $w \in V - \{v\}$, segundo a ordenação da etapa 1 e da ordem em que foi incluido no conjunto V, e faça $v := w$. Volte à etapa 2.

4. Se $V \neq V(G)$ e não tiver vértice pertencente a $V(G) - V$ adjacente a v, escolha o menor vértice x pertencente a $V - \{v\}$, segundo a ordenação da etapa 1 e da ordem em que foi incluido no conjunto V, e faça $v := x$. Volte à etapa 3.

3.6 Exercícios (Grafos)

3.1 *Desenhe os grafos dos itens **(a)** a **(d)** abaixo.*

(a) $V(K_{3,4}) = \{v_1, v_2, v_3, w_1, w_2, w_3, w_4\}$,

$E(K_{3,4}) = \{v_1w_1, v_1w_2, v_1w_3, v_1w_4, v_2w_1, v_2w_2, v_2w_3, v_2w_4, v_3w_1, v_3w_2, v_3w_3, v_3w_4\}$

(b) $V(G) = \{v_1, v_2, v_3, v_4, v_5\}, E(G) = \{v_1v_1, v_2v_3, v_2v_4, v_4v_3, v_4v_3\}$

(c) $V(H) = \{w_1, w_2, w_3, w_4, w_5, w_6\}, E(H) = \{w_1w_2, w_2w_3, w_3w_4, w_4w_5, w_5w_6, w_6w_1\}$

(d) $V(G_2) = \{v_1, v_2, v_3, v_4, v_5, v_6\}, E(G_1) = \{v_1v_2, v_1v_2, v_1v_3, v_3v_4, v_2v_5, v_4v_6, v_3v_2, v_3v_3\}$

3.2 *Quais grafos da questão **1** são simples?*

3.3 *Qual o grau de cada vértice dos grafos da questão **1**?*

3.4 *Quais grafos da questão **1** são conexos?*

3.5 *Escreva os grafos da questão **1** na forma matricial.*

3.6 *Desenhe o grafo G_2, descrito na forma matricial, abaixo. Depois, descreva os elementos dos conjuntos $V(G_2)$ e $E(G_2)$. O grafo G_2 é simples?*

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 3 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 3 & 1 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{bmatrix}$$

3.7 *Desenhe o grafo G_3, descrito na forma matricial, abaixo. Depois, descreva os elementos dos conjuntos $V(G_3)$ e $E(G_3)$. O grafo G_3 é simples?*

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 & 0 \end{bmatrix}$$

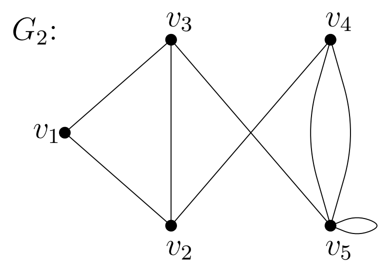
3.8 *Para o grafo G_2 a seguir:*

(a) *Determine a matriz de adjacência de G_2.*

(b) *O grafo G_2 é simples? Justifique.*

(c) *Exiba um circuito do grafo G_2.*

(d) *O grafo G_2 é conexo? Justifique.*



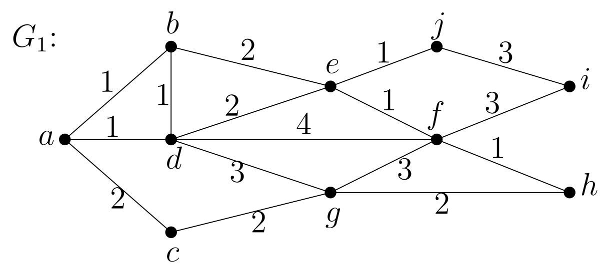
3.9 *Para o grafo G_1 a seguir:*

(a) *Determine o grau dos vértices a, e e h.*

(b) *Determine uma árvore geradora de G_1 com custo mínimo.*

(c) *Desenhe o subgrafo H_1 onde $V(H_1) = \{e, f, i, j, h, g\}$ e $E(H_1) = \{ej, ef, fi, fg, fh, gh\}$.*

(d) *Calcule o custo do subgrafo H_1.*



3.10 *Considere o grafo G_3 abaixo.*

(a) *Desenhe o subgrafo H de G_3, onde*

$V(H) = \{v_3, v_4, v_5, v_6, v_7\}$ *e* $E(H) = \{v_5v_4, v_4v_6, v_3v_7, v_3v_4\}$.

(b) *Determine a matriz de adjacência do subgrafo H.*

(c) *O subgrafo H é uma árvore? Justifique.*

(d) *Começando pelo vértice v_7, determine uma árvore geradora T de G_3 com custo mínimo.*

(e) *Calcule o custo da árvore T determinada no item* (a).