Básicos.

E 1.20 Mostre que todo grafo tem um número par de vértices de grau ímpar.

E 1.21 Mostre que todo grafo com dois ou mais vértices tem pelo menos dois vértices de mesmo grau.

E 1.26 Mostre que se G é um grafo com $\delta(G) > 0$ e $m(G) < n(G)$ então G tem pelo menos dois vértices de grau 1.

Cortes

Dado um conjunto $X \subseteq V(G)$, o corte definido por X, representado por $\nabla(X)$, é o conjunto das arestas que ligam os vértices em X aos vértices fora de X. Formalmente, $\nabla(X) = \{ \{u,v\} : \{u,v\} \in A(G) \text{ e } u \in X \text{ e } v \notin X \}$.

M 1.1 Mostre que $m(G) = m(G[X]) + m(G-X) + |\nabla_G(X)|$.

Ciclos e árvores

E 1.31 Mostre que todo corte de um circuito tem cardinalidade par. Isto é, mostre que, para qualquer conjunto X de vértices de um circuito O , o corte $\nabla_O(X)$ tem cardinalidade par.

E 1.33 Uma floresta é um grafo sem circuitos. Mostre que um grafo G é uma floresta se e somente se cada uma de suas arestas é um corte, ou seja, para cada aresta a existe um subconjunto X de V(G) tal que $\nabla(X) = \{a\}$.

Conexidade

E 1.38 Seja G um grafo tal que $\Delta(G) \leq 2$. Mostre que cada componente de G é um caminho ou um circuito.

E 1.39 Seja G um grafo com $\delta(G) \geq \lfloor n(G)/2 \rfloor$. 11 Mostre que G é conexo.

E 1.40 Seja G um grafo com $\delta(G) \geq 3$. Mostre que G tem um circuito par.

E 1.41 Mostre que todo grafo G satisfaz a desigualdade $m(G) \geq n(G) - c(G)$, na qual $c(G)$ é o número de componentes de G.

E 1.42 Uma árvore é uma floresta (veja exercício 1.33) conexa. (Os grafos discutidos no exemplo 1.9 são árvores.) Mostre que um grafo conexo G é uma floresta se e somente se, para cada aresta a, o grafo $G - a$ não é conexo.

Caminhos mínimos

M 1.2 Suponha que o custo de cada arco de um grafo orientado G seja sempre unitário, ou seja c(a) = 1 para toda a .in. A(G). Qual é o melhor algoritmo para encontrar todos os caminhos de custo mínimo partindo de um mesmo vértice original dado (s) para todos os demais vértices do grafo? Qual é a complexidade do algoritmo?

M 1.3 E se o custo de cada arco do exercício 1.2 fosse sempre igual a uma constante positiva, porém não necessariamente 1. O mesmo algoritmo funcionaria? Por que?

M 1.4 Suponha que você executou um algoritmo de caminhos mínimos como Dijkstra ou Bellman-Ford em um grafo G e origem s. O algoritmo encontrou a árvore de caminhos mínimos e os rótulos de custos dos caminhos para cada vértice do grafo. Ao analisar o resultado, percebeu-se que um arco {u,v} não pertence a árvore. Suponha que desejamos que a árvore contenha o arco mencionado e que seja possível reduzir o custo do arco no grafo. Qual seria a menor redução necessária para que o arco pertença a algum caminho de custo mínimo?

M 1.5 Suponha que um grafo não contém ciclos negativos e possui ao menos três vértices distintos: a, b e c. Mostre que, se existe um caminho de a para b com custo $c1$ e um caminho de b para c com custo $c2$, então existe um caminho de a para c com custo menor ou igual a $c1+c2$. Mostre que não podemos garantir que exista um caminho de custo $c1+c2$ de a para c. Porque precisamos da hipótese que não há ciclos negativos? Mostre um grafo com ciclos negativos que a proposição apresentada falha.

Coloração de vértices

Seja H um subgrafo de G, mostre que: $\omega(G) \geq \omega(H)$, mas não há relação entre $\alpha(G)$ e $\alpha(H)$. Mostre que $\chi(G) \geq \chi(H)$.

Seja H um subgrafo induzido de G, mostre que: $\omega(G) \geq \omega(H)$ e $\alpha(G) \geq \alpha(H)$. Mostre que $\chi(G) \geq \chi(H)$.

Mostre que se v é um vértice de G tal que $g(v) < \chi(G) - 1$ e $H = G - v$, então $\chi(G) = \chi(H)$.

Sejam G1 e G2 dois grafos com conjuntos disjuntos de vértices. Define-se a operação de junção $H = G1 + G2$, de forma que: $V(H) = V(G1) \cup V(G2)$, $A(H) = A(G1) \cup A(G2) \cup \{ \{u,v\} : u \in V(G1) \text{ e } v \in V(G2) \}$.
Prove que $\chi(H) = \chi(G1) + \chi(G2)$, $\omega(H) = \omega(G1) + \omega(G2)$ e $\alpha(H) = \text{máx } \{\alpha(G1), \alpha(G2)\}$.