Resumo Aula 5 - Algoritmos em Grafos

Vinicius Bernardo Gabriel

14 de Março de 2020

1 Lema sobre articulações e pontes

Lema. *Seja $G = (V, E)$ um grafo com $n >= 2$ (número de vértices), então:*

1. *$v \in V(G)$ é uma articulação $\Leftrightarrow$ v está em todos os caminhos entre dois vértices u e w de G.*
2. *$e \in E(G)$ é uma ponte $\Leftrightarrow$ e é o único caminho entre dois vértices u e v de G.*

Prova (1). Ida. Suponha que v é uma articulação. Por definição, $G-v$ é desconexo, e portanto, existem dois vértices u e w tal que não existe caminho entre eles. Mas, como G pode ter múltiplos caminhos entre u e w e $G - v$ não tem mais nenhum caminho entre u e w, isso implica que v estava em todos os caminhos entre u e w. Volta. Suponha que v está em todos os caminhos entre dois vértices u e w, logo, $G - v$ é desconexo, pois não teremos mais caminho entre u e w (removemos v), por definição, v é uma articulação, pois $G - v$ é desconexo. ■

Prova (2). Ida. Suponha que e é uma ponte. Por definição, $G - e$ é desconexo, e portanto, existem dois vértices u e v sem caminho entre eles. Porém, em G poderiam existir multiplos caminhos entre u e v, mas como $G-e$ é desconexo por definição, isso implica que e era o único caminho entre u e v em G (caso contrário, $G - e$ ainda seria conexo, contradizendo a definição de ponte). Volta. Suponha que e é o único caminho entre dois vértices u e v, logo, $G - e$ é desconexo, portanto, pela definição de ponte, e é uma ponte, pois $G - e$ é desconexo. $\blacksquare$

2 Digrafos (Grafos dirigidos, orientados, direcionado)

Um **digrafo** $D = (V, A)$ é definido por um conjunto $V(D)$ de vértices e um conjunto $A(D)$ de **pares ordenados**[1] de vértices em $V(D)$, os elementos de $A(D)$ são comumente chamados de **arcos** ou de **arestas**. Um **arco é representado graficamente por uma seta**, onde a cauda (ou inicio da seta) é a primeira coordenada e a cabeça da seta é sua primeira coordenada.

[1]Quer dizer que a ordem importa: (a, b) é diferente de (b, a)

2.1 Grau de entrada

O grau de entrada de um vértice $v \in D$, denotado por $grau^{-}(v)$ é o número de arcos tais que v é a cabeça deles, ou seja, onde v **é a segunda coordenada**.

2.2 Grau de saída

O grau de saída de um vértice $v \in D$, denotado por $grau^{+}(v)$ é o número de arcos tais que v é a cauda deles, ou seja, onde v **é a primeira coordenada**.

2.3 Grau (como de um grafo não orientado)

Note que $grau(v) = grau^+(v) + grau^-(v)$.

2.4 Grau máximo Δ^+ e Δ^- de D

Definimos $\Delta^+(D)$ como sendo o maior grau de saída de D e $\Delta^-(D)$ como sendo o maior grau de entrada de D.

2.5 Fontes e sumidouros

Um vértice v em um digrafo D é chamado de **fonte** se seu grau de entrada é 0 ($grau^-(v) = 0$) ou de **sumidouro** se seu grau de saída é 0 ($grau^+(v) = 0$).

2.6 Alcance

Se existe um caminho (orientado) entre um vértice $v \in D$ e qualquer outro vértice $u \in D$, dizemos que v alcança todos os vértices de D.

3 Grafo subjacente

Dado um digrafo D, podemos obter o **grafo subjacente** de D. Esse grafo é obtido pela remoção da orientação dos arcos de D, e em seguida pela remoção das arestas multiplas e laços, caso existam.

4 Digrafos Conexos, Fortemente Conexos e Unilateralmente Conexos

4.1 Digrafo Conexo

Dizemos que um digrafo D é **conexo** se o seu **grafo subjacente é conexo**.

4.2 Digrafo Fortemente Conexo

Dizemos que um digrafo D é **fortemente conexo** se $\forall u, v \in V(D)$, existe um **caminho direcionado** de u para v e de v para u.

4.3 Digrafo Unilateralmente Conexo

Dizemos que um digrafo D é **unilateralmente conexo** se ele não for fortemente conexo, mas $\forall u, v \in V(D)$, existe um caminho de v para u ou de u para v.

5 Componentes fortemente conexas

Uma componente fortemente conexa de D é um **subgrafo maximal** de D **fortemente conexo**.

6 DAG - Digrafos Acíclicos

Um DAG (Directed Acyclic Graph) é um digrafo sem ciclos direcionados.

7 Grafos Ponderados

Um grafo é ponderado se ele possui alguma função ou um peso (capacidade) associada a cada aresta ou vértice do grafo.

8 Árvores Direcionadas

Uma árvore enraizada direcionada (arborência) em r significa que D é uma árvore enraizada em r e:

$$d^-(r) = 0$$
$$d^-(v) = 1, \forall v \neq r$$

9 Representação de Grafos em Computador

9.1 Matriz de Adjacência[1]

A **matriz de adjacência de um grafo** $G = (V, E)$ é uma matriz R com dimensões nxn, onde:

- Se não existe aresta entre o par de vértices (V_i, V_j), então o elemento R_{ij} da matriz de adjacência é igual a 0 ($R_{ij} = 0$)
- Caso contrário, isto é, existe alguma aresta entre (V_i, V_j), então o elemento R_{ij} é igual ao número de arestas entre o par (V_i, V_j). $\forall R_{ij} \neq 0, \exists e = (V_i, V_j) \in E(G)$.
- Se o grafo G for ponderado, então usamos $-\infty$ para indicar que não existe aresta entre um par de vértices.

[1]Lembre-se que um vértice v_i é adjacente (ou vizinho) a outro vértice v_j se existe alguma aresta (v_i, v_j) ou (v_j, v_i)

A matriz de adjacências possui limitações quando temos grafos ponderados com multiplas arestas, nesse caso, pode ser mais interessante usar outra maneira para representar o grafo. A **complexidade de espaço** da matriz de adjacências é $O(n^2)$, pois temos n linhas com vértices e n colunas na matriz.

Exemplo de Matriz de Adjacência

Com um grafo $G = (V, E)$, $V = [a, b, c]$ e $E = [(a, b), (b, c), (c, a)]$

$$\begin{array}{c c} & \begin{array}{ccc} a & b & c \end{array} \\ \begin{array}{c} a \\ b \\ c \end{array} & \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{bmatrix} \end{array}$$

De forma resumida, cada linha é respectiva a um vértice v_i, e cada coluna é para um outro vértice v_j (que também pode ser o próprio v_i). O elemento R_{ij} da matriz diz se v_i é adjacente a v_j, ou seja, se existe uma aresta (v_i, v_j) ou (v_j, v_i). Analisando o exemplo, vemos que o vértice a é adjacente a b e c, portanto existem arestas (a, b) e (c, a). b é adjacente a a e c, portanto temos arestas (b, a), (b, c). E por fim, c é adjacente a a e b, portanto, temos arestas (a, c) e (b, c). Note que temos o grafo G que desejavamos representar, e também seria possível representar um grafo com arestas repetidas e com laços, bastando alterar um pouco a maneira que interpretamos as informações. Um fato interessante, é que a **matriz de adjacência é uma matriz simétrica** (para grafos não direcionados). Isso fica claro porque em um grafo não direcionado, a relação criada por uma aresta entre dois vértices é **simétrica**, ou seja, se a é adjacente a b, então b é adjacente a a.

9.2 Matriz de Incidência

A **matriz de incidência** é similar a matriz de adjacências, temos uma linha para cada vértice (portanto n linhas) e uma coluna para cada aresta (m colunas). Formamos os elementos da matriz da seguinte maneira:

- Se o vértice v_i não incide[1] na aresta e_j, então o elemento $R_{ij} = 0$. Isto é, o elemento na linha i e coluna j é zero.
- Caso contrário, ou seja, v_i incide em e_j, então o elemento na linha i e coluna j é 1, $R_{ij} = 1$.

[1]Um vértice v incide em uma aresta e se v está em alguma das extremidades de e, ou seja, $e = (v, u)$ ou $e = (u, v)$.

Usando o exemplo anterior, com um grafo $G = (V, E)$,
$V = [a, b, c]$, $E = [e_1 = (a, b), e_2 = (b, c), e_3 = (c, a)]$

$$\begin{array}{c c} & \begin{matrix} e_1 & e_2 & e_3 \end{matrix} \\ \begin{matrix} a \\ b \\ c \end{matrix} & \begin{bmatrix} 1 & 0 & 1 \\ 1 & 1 & 0 \\ 0 & 1 & 1 \end{bmatrix} \end{array}$$

É interessante notar que a soma de ums em cada coluna sempre deve ser 2, pois cada aresta incide exatamente em dois vértices. E também que a soma dos "ums" em cada linha é exatamente o grau do vértice correspondente à linha. Novamente, temos que o vértice a incide em e_1 e e_3, b incide em e_1 e e_2 e por fim, c incide em e_2 e e_3. Note que diferentemente da matriz de adjacências, essa matriz não necessariamente será quadrada (e portanto, também não será simétrica). A complexidade em espaço é da ordem de $O(nm) = O(n^3)$[2] no pior caso. Mas, em geral, essa matriz geralmente vai ser melhor que a matriz de adjacências, pois tipicamente temos mais vértices que arestas, logo $m \leq n$. Essa matriz só é um problema quando a matriz vai se tornando mais densa[3], pois m tende a $O(n^2)$, fazendo com que acomplexidade nesse caso seja $O(n^3)$. Com essa matriz, também é possível representar grafos com arestas múltiplas e pesos, sendo possível usar a própria matriz para guardar os pesos ou um vetor auxiliar pra guardar o peso de cada aresta.

[2]Quando a matriz é completa, temos $m = \binom{n}{2} = \frac{n(n-1)}{2} = \frac{n^2-n}{2}$ arestas, portanto, $m \simeq O(n^2)$ quando a matriz é completa.
[3]O grafo é denso quando tem muitas arestas, ou seja, quando o número de arestas se aproxima de n^2, caso contrário, podemos dizer que o grafo é esparso.

9.3 Lista de Adjacências

Dado um grafo $G = (V, E)$, a sua **lista de adjacências** é formada por um **vetor** H de n elementos, onde cada elemento H_i do vetor corresponde ao i-esimo vértice (V_i) de G, e esse elemento H_i aponta para uma lista de vértices adjacentes ao vértice V_i de G.

Representando o mesmo grafo de antes $G = (V, E)$, $V = [a, b, c]$, $E = [(a, b), (b, c), (c, a)]$

$$
\begin{array}{l}
a \longmapsto b \rightarrow \ c \rightarrow \ null \\
b \longmapsto a \rightarrow \ c \rightarrow \ null \\
c \longmapsto a \rightarrow \ b \rightarrow \ null
\end{array}
$$

Com a lista de adjacências, é possível representar qualquer grafo, inclusive os grafos ponderados com arestas multiplas, para as arestas multiplas, basta repetir os vértices na lista de vértices adjacentes, e a questão do peso, é possível criar novos campos para representar os pesos. Essa estrutura é em geral a estrutura mais leve para representar os grafos, tendo complexidade de espaço no pior caso de $O(n+m) = O(n+n^2) = O(n^2)$. Ela é melhor que as outras duas porque em geral, m não é tão grande (menor que n^2). Se m for realmente grande (na ordem de n^2 ou superior), talvez outra estrutura seja melhor. É interessante observar também que, embora estejam descritas dessa forma, é possível adaptar cada estrutura de forma que se adeque melhor ao problema em questão, por exemplo, é possível ao invés de repetir vértices para representar arestas múltiplas, adicionar um campo indicando quantas vezes a aresta ocorre, entre outras modificações que podemos fazer nas outras estruturas.