CPS740 - Lista 4

Pedro Maciel Xavier
116023847

8 de setembro de 2020

Questão 1.:

Seguindo as especificações do grafo obtemos:

$$V = \{(2,3),(2,5),(2,7),(3,2),(3,5),(3,7),\\ (5,2),(5,3),(5,7),(7,2),(7,3),(7,5)\}$$

$$E_1 = \{((2,3),(3,2)),((2,5),(5,2)),((2,7),(7,2)),\\ ((3,5),(5,3)),((3,7),(7,3)),((5,7),(7,5))\}$$

$$E_2 = \{((2,3),(2,5)),\ ((2,3),(2,7)),\ ((2,5),(2,7)),\\ ((3,2),(3,5)),\ ((3,2),(3,7)),\ ((3,5),(3,7)),\\ ((5,2),(5,3)),\ ((5,2),(5,7)),\ ((5,3),(5,7)),\\ ((7,2),(7,3)),\ ((7,2),(7,5)),\ ((7,3),(7,5))\}$$

a) Desenhamos $G(V, E_1 \cup E_2)$ com os respectivos pesos indicados em cada uma das arestas. As arestas pertencentes a E_1 foram coloridas de azul, enquanto as de E_2 estão vestidas de vermelho.

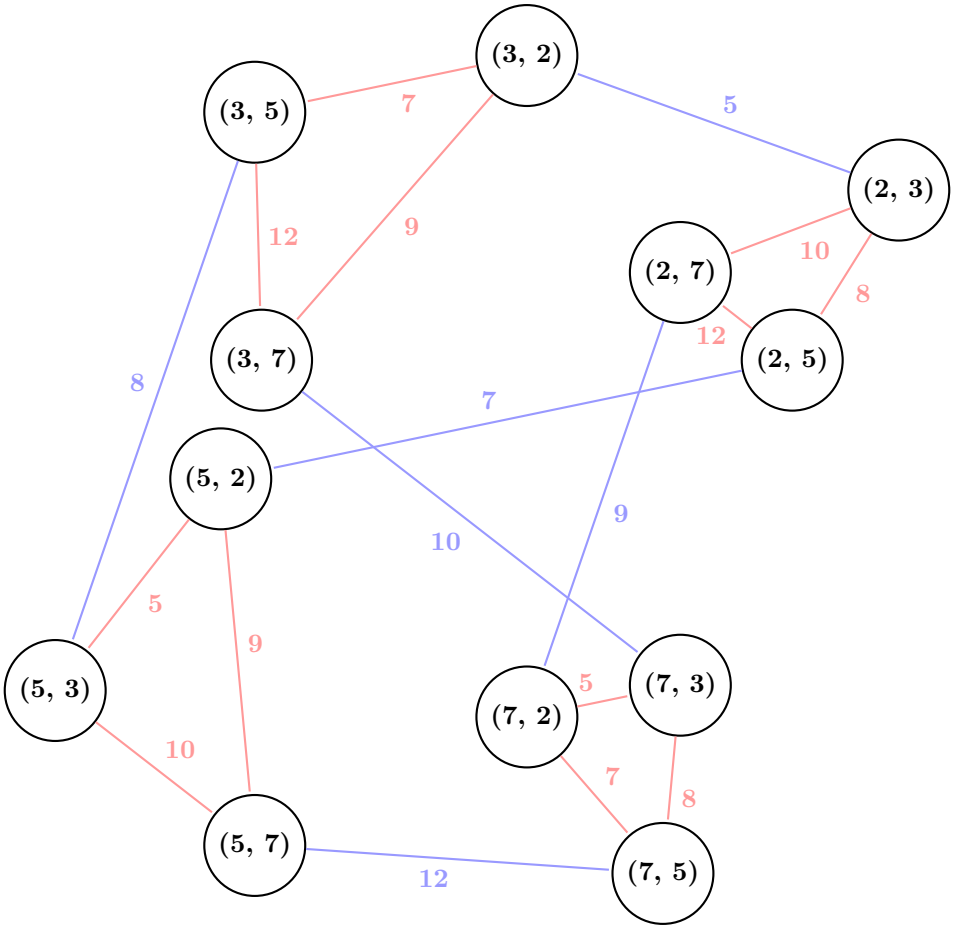
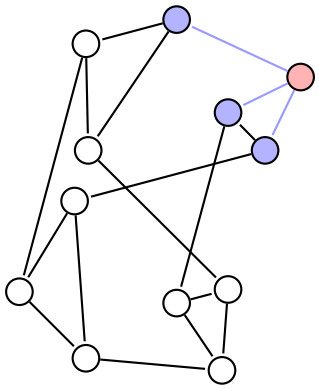
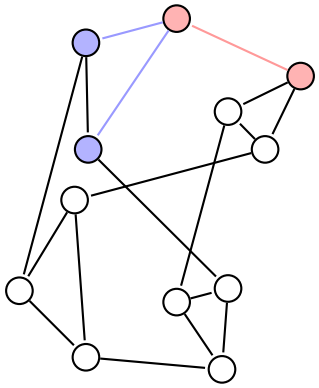


Figura 1: O Grafo $G(V, E_1 \cup E_2)$

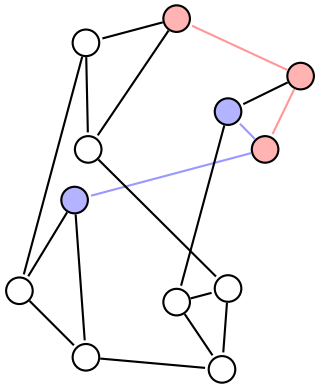
b) Começando no vértice $(2,3)$, vejamos a execução do algoritmo de *Djkstra*, com a miniatura do grafo ao lado do vetor de distâncias.



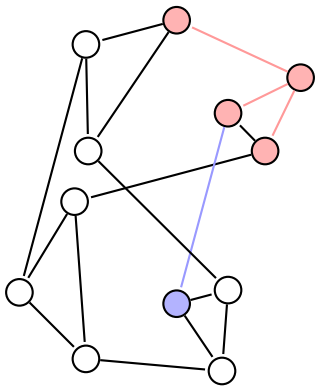
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	∞
$(3,7)$	∞
$(5,2)$	∞
$(5,3)$	∞
$(5,7)$	∞
$(7,2)$	∞
$(7,3)$	∞
$(7,5)$	∞



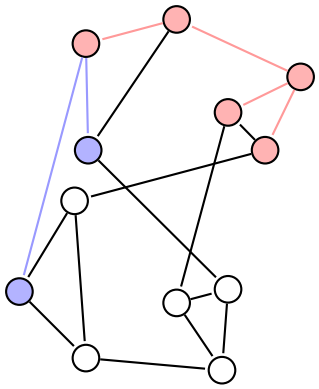
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	∞
$(5,3)$	∞
$(5,7)$	∞
$(7,2)$	∞
$(7,3)$	∞
$(7,5)$	∞



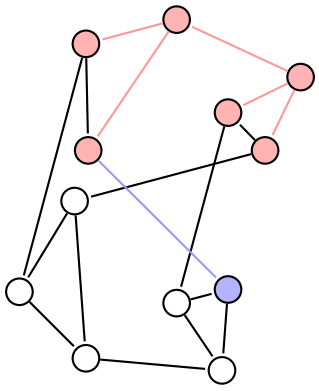
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	∞
$(5,7)$	∞
$(7,2)$	∞
$(7,3)$	∞
$(7,5)$	∞



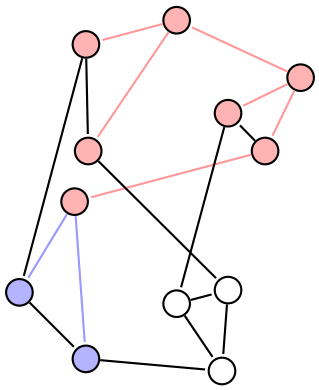
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	∞
$(5,7)$	∞
$(7,2)$	19
$(7,3)$	∞
$(7,5)$	∞



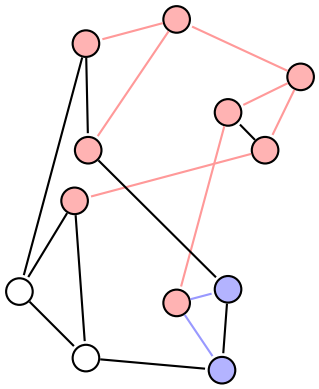
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	∞
$(7,2)$	19
$(7,3)$	∞
$(7,5)$	∞



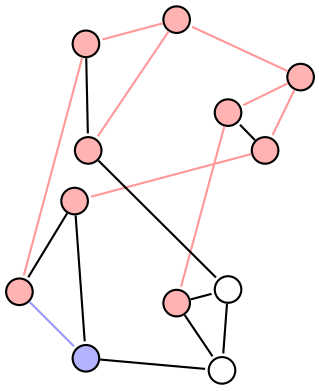
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	∞
$(7,2)$	19
$(7,3)$	24
$(7,5)$	∞



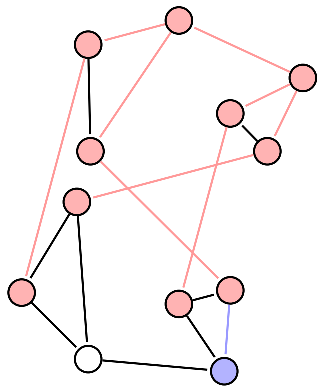
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	24
$(7,2)$	19
$(7,3)$	24
$(7,5)$	∞



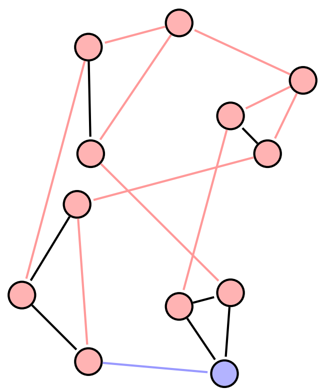
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	24
$(7,2)$	19
$(7,3)$	24
$(7,5)$	26



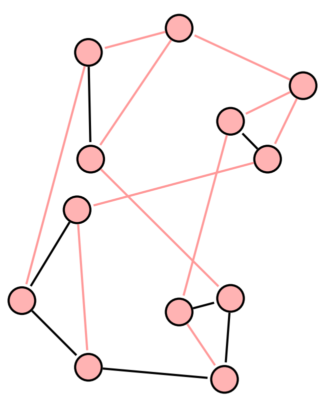
v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	24
$(7,2)$	19
$(7,3)$	24
$(7,5)$	26



v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	24
$(7,2)$	19
$(7,3)$	24
$(7,5)$	26



v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	24
$(7,2)$	19
$(7,3)$	24
$(7,5)$	26



v	$d(u,v)$
$(2,3)$	0
$(2,5)$	8
$(2,7)$	10
$(3,2)$	5
$(3,5)$	12
$(3,7)$	14
$(5,2)$	15
$(5,3)$	20
$(5,7)$	24
$(7,2)$	19
$(7,3)$	24
$(7,5)$	26

Questão 2.:

Algoritmo 1.: Floyd-Warshall++

```
def floyd_warshall(G(V, E), P[n][n]):
    // `G(V, E)` um grafo
    // `P[n][n]` a matriz de pesos das arestas de G
    seja n ← |V|

    seja dist[n][n] ← ∞
    seja kmin[n][n] ← nulo

    para cada [u, v] em E:
        dist[u][v] ← P[u][v]
        kmin[u][v] ← v

    para cada v em V:
        dist[v][v] ← 0
        kmin[v][v] ← v

    para k de 1 até n:
        para i de 1 até n:
            para j de 1 até n:
                se dist[k][k] < 0:
                    // Achamos um ciclo negativo
                    retorna nulo
                se dist[i][j] > dist[i][k] + dist[k][j]:
                    // Preferimos ir de `i` para `k`
                    // do que de `i` para `j`
                    dist[i][j] ← dist[i][k] + dist[k][j]
                    kmin[i][j] ← kmin[i][k]

    seja caminhos[n][n]

    para cada [u, v] em V × V:
        se kmin[u][v] = nulo:
            caminhos[u][v] ← nulo
        senão:
            caminhos[u][v] ← lista([u])
            enquanto u != v:
                u ← kmin[u][v]
                caminhos[u][v].inserir(u)

    retorna caminhos
```

Questão 3.:

1 .: O algoritmo de *Dijkstra* sempre obtém o caminho mais curto caso o grafo possua arestas com pesos negativos. **Falso**

Seja $G(V, E)$ com $V = \{x, y, z\}$ e $E = \{(x, y), (x, z), (y, z)\}$. Suponha que $P(x, y) = a$, $P(x, z) = b$ e $P(y, z) = -b$ com $0 < a < b$.

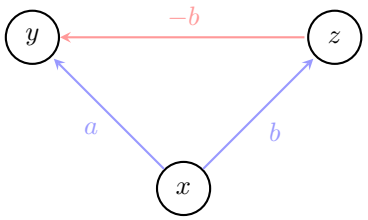


Figura 2: Grafo genérico onde o algoritmo de *Djkstra* falha.

O Algoritmo de *Djkstra* consideraria (x, y) o menor caminho entre x e y, negligenciando a existência de (x, z, y), cujo custo é menor.

2 .: Sejam M e M' dois emparelhamentos perfeitos de G. Seja H o subgrafo induzido pela diferença simétrica de M e M'. Então, toda componente conexa de H é um ciclo ou caminho. **Verdadeiro**

Demonstração. Vamos desconsiderar o caso trivial onde $M = M'$, pois bastaria tomar $G = K_2$ para obter um contra-exemplo ($M \triangle M' = \emptyset$). Dito isto, vamos supor, por absurdo, que uma das componentes conexas de H não é um ciclo ou caminho, isto é, possui ao menos um vértice $w \in H, \text{grau}(w) \geq 3$.

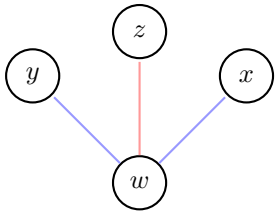


Figura 3: Componente conexa hipotética em H

Vamos supor que w esteja conectado aos vértices $x, y, z \in H$ e que $\{(w, x), (w, y), (w, z)\} \subseteq M \triangle M'$. Como $A \triangle B = (A - B) \cup (B - A)$, então $e \in A \triangle B \implies e \notin A \cap B$. Como uma aresta de H não pode estar em M e M' ao mesmo tempo, temos, pelo Princípio da Gaiola de Pombos (*Pigeonhole*), que um dos dois conjuntos deve possuir ao menos duas arestas. No entanto, as três arestas são incidentes em w. Isto não é possível, uma vez que M e M' são emparelhamentos. ■

Questão 4.:

Seja $G(V, E)$ um grafo simples e M um emparelhamento máximo de G.

Questão 5.:

Suposição. *Não é possível cobrir um tabuleiro de* 100×100 *utilizando retângulos* 2×1 *após a retirada de dois cantos opostos.*

Demonstração. Imaginemos que o tabuleiro é colorido como se fosse para jogar xadrez ou damas:

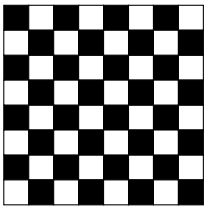


Figura 4: Tabuleiro de Xadrez

Ao posicionar um retângulo 2×1 sobre o tabuleiro, este ocupará sempre uma casa clara e uma casa escura. No entanto, ao removermos cantos opostos do tabuleiro, isto é, casas de uma mesma diagonal principal, estaremos retirando duas casas de mesma cor. Desta forma, tendo um tabuleiro de tamanho $n \times n, n \geq 2$ após a remoção dos cantos, podemos posicionar até $n^2/2 - 2$ retângulos. n é necessariamente um número par. Ao tentar incluir o que seria o último retângulo do tabuleiro ferido, nos deparamos com duas casas de mesma cor. Posicionar a peça violaria a necessidade de preencher casas de cores diferentes. ■

Referências

[1] Jayme Luiz Szwarcfiter, **Teoria Computacional de Grafos**, 1ª edição, Rio de Janeiro, 2018.

[2] Brunold, **Aplicativo de Mensagens Telegram**, 2020.