1. Seja $T = (V, E)$ uma árvore. Mostre que todos os vértices de T têm grau ímpar se, e somente se, para toda aresta $e \in E$, ambas as componentes de $T - e$ têm um número ímpar de vértices.

($\Rightarrow$) Todos os vértices de T têm grau ímpar $\Rightarrow$ $\forall e \in E$, ambas componentes de $T-e$ têm número ímpar de vértices

Como T é uma árvore, ela possui $|V|-1$ arestas e $|E| = |V|-1$. Sabemos que $\sum \delta(v) = 2|E|$. Como todos os vértices têm grau ímpar, então como $\sum \delta(v) = 2|E|$, há uma quantidade par de vértices. Assim, temos uma quantidade ímpar de arestas.

Portanto, cada componente conexa de $T-e$ (duas pois T é árvore), para toda $e \in E$, temos apenas um vértice de grau par. Como $\sum \delta(v) = 2|E|$, então para cada componente teremos uma quantidade par de vértices de grau ímpar e o vértice de grau de par. Logo, ambas componentes conexas terão número ímpar de vértices.

($\Rightarrow$) $\forall e \in E$, ambas componentes de $T-e$ têm número ímpar de vértices $\Rightarrow$ Todos os vértices de T têm grau ímpar

Se ambas componentes têm número ímpar de vértices para toda $e \in E$, então o número de vértices é par.

Seja $v \in V$ arbitrário. Seja e aresta incidente em v. Assim, no grafo $T-e$, temos duas componentes conexas, uma que contém v e outra que não, ambas com número ímpar de vértices.

Retirando todas as arestas incidentes em v, obtemos $\delta(v)$ componentes conexas disjuntas sem

o vértice v todas com número ímpar de vértices. Como o número de vértices é par, então $\delta(v)$ é ímpar.
Como v foi tomado arbitrariamente, então $\forall v \in V$, $\delta(v)$ é ímpar.

2. Mostre que todo grafo k-cromático contém pelo menos $\binom{k}{2}$ arestas.

Grafo K-cromático: O número mínimo de coloração por vértices é K.
Seja $c_1, \ldots, c_i, c_j, \ldots, c_K$ uma K-coloração válida.
Suponha que $\exists c_i, c_j$ tais que nenhuma aresta conecta vértices coloridos com tais cores, então, podemos colorir o vértices da cor c_i com c_j e G é $(K-1)$-colorível, o que é um absurdo.
Assim, deve existir pelo menos uma aresta ligando vértices coloridos com c_i, c_j.
Portanto, como existem $\binom{K}{2}$ pares distintos de cores, existem, ao menos uma aresta para cada par distinto. Assim, G possui $\binom{K}{2}$ arestas.

perfeito.

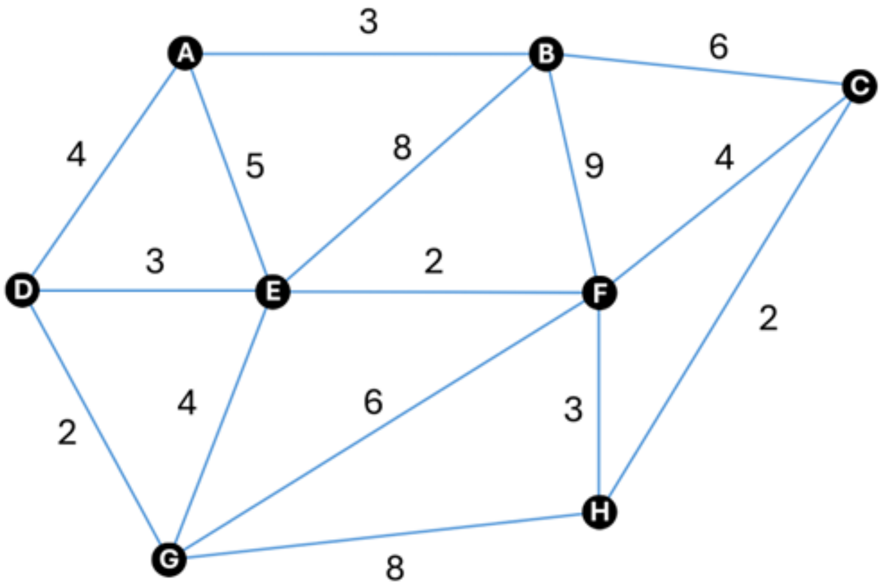
3. Mostre que toda árvore possui, no máximo, 1 matching [redacted].

Matching perfeito: matching que contém todos os vértices do grafo.

Uma árvore é um grafo 2-cromático (excluindo a árvore trivial). Além disso, um grafo é bipartido se, e só se, é 2-cromático (conexo, não trivial, sem laços). Logo, toda árvore é bipartida.
Portanto, se o número de vértices da árvore for ímpar, não conseguimos montar nenhum matching perfeito, pois não conseguimos relacionar todos os vértices.
Vamos provar por indução que para um número par de vértices, uma árvore tem 1 matching perfeito.

Caso base: $n=2$ (n é o número de vértices). Trivial
Hipótese: Toda árvore com $k=n$ (par) vértices tem um matching perfeito.
Queremos verificar se a hipótese vale para $k=n+2$.
Sejam u uma folha e v o vértice adjacente a u. Assim, no matching, u e v estão conectados.
Considerando o grafo $T-\{u,v\}$, toda componente conexa, por hipótese tem um matching perfeito.
Como o número de matchings perfeitos é o produto do número de matchings perfeitos, então T possui um matching perfeito.
Como uma árvore possui 0 ou 1 matching perfeito, então o número máximo é 1.

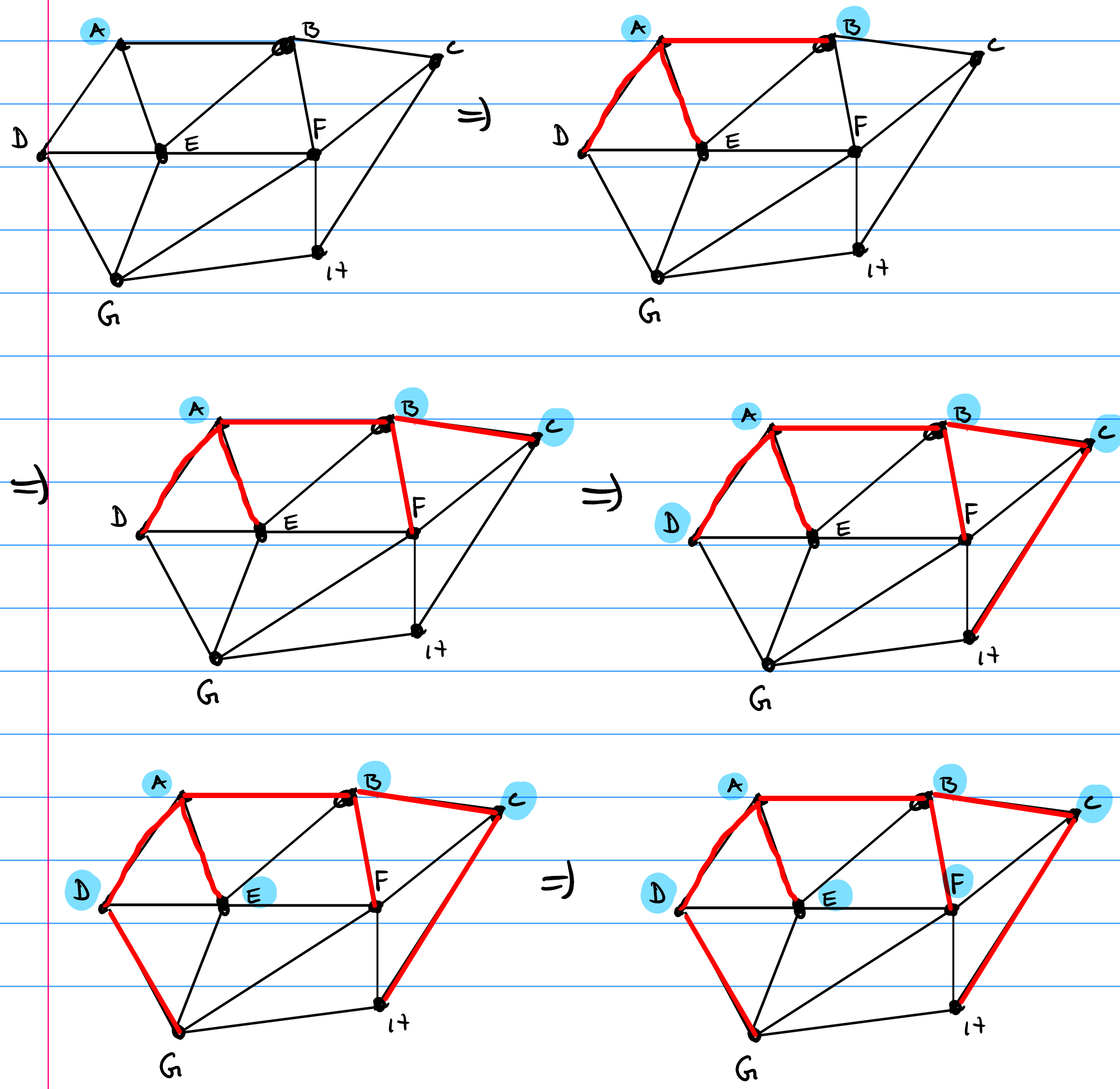
4. No grafo a seguir, encontre uma árvore geradora

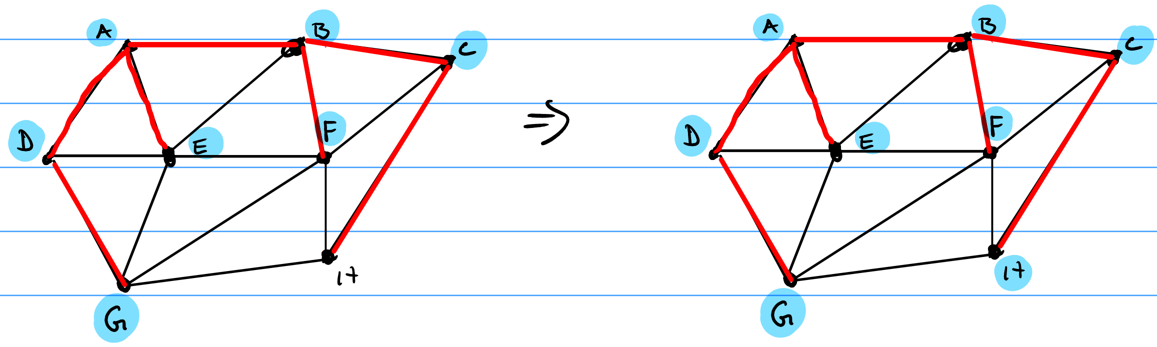


(a) sem levar em conta os pesos das arestas. Detalhe as iterações do algortimo.

(b) minimizando a soma dos pesos das arestas. Detalhe as iterações do algoritmo.

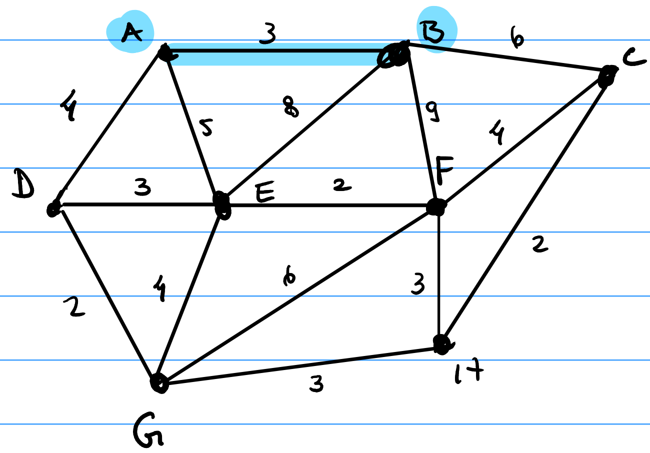
a) usando um BFS começando em A:



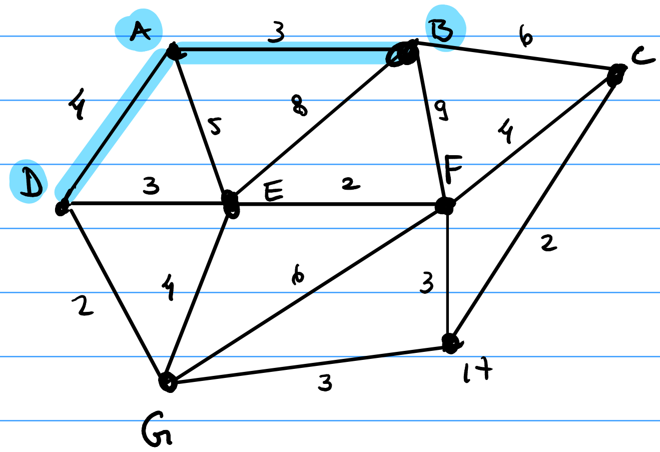


Ordem: {A, B, C, D, E, F, G, H}.

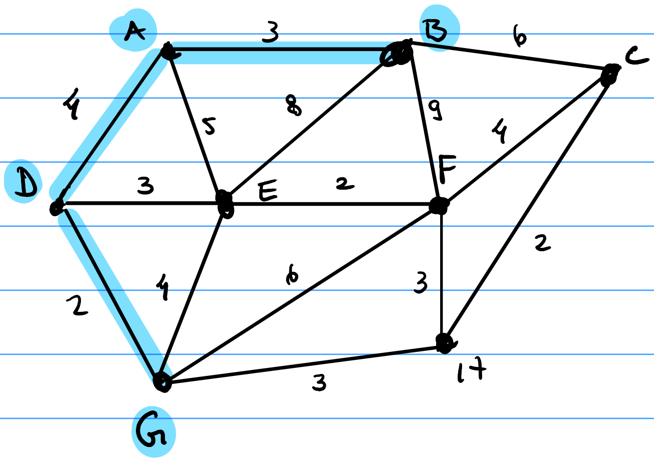
b) Usando o algoritmo de Prim, inicializando em A:



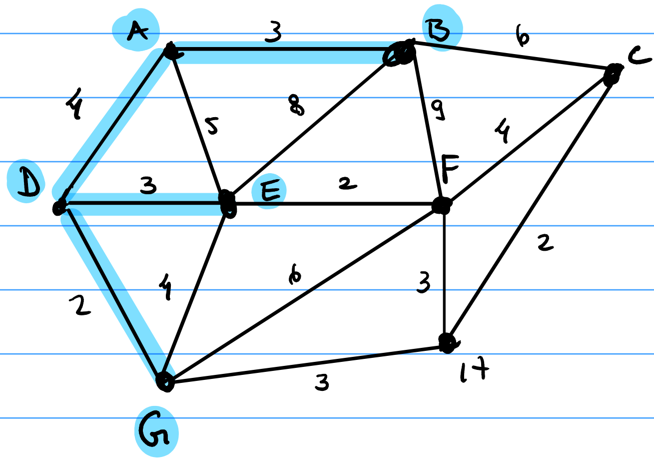
Arestas	Peso
AD	4
AB	3
AE	5



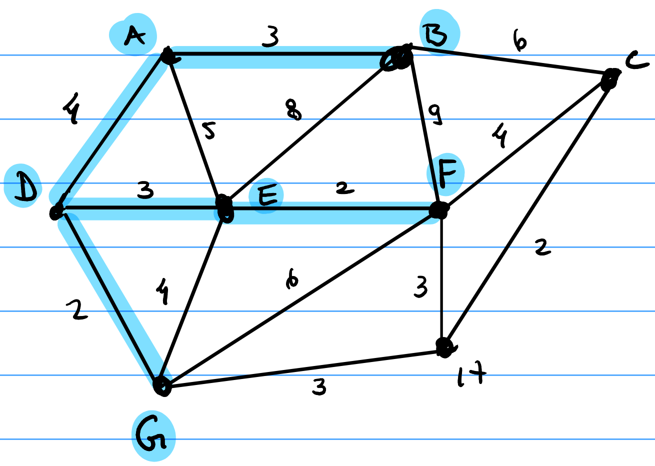
Arestas	Peso
AD	4
AE	5
BC	6
BF	9
BE	8



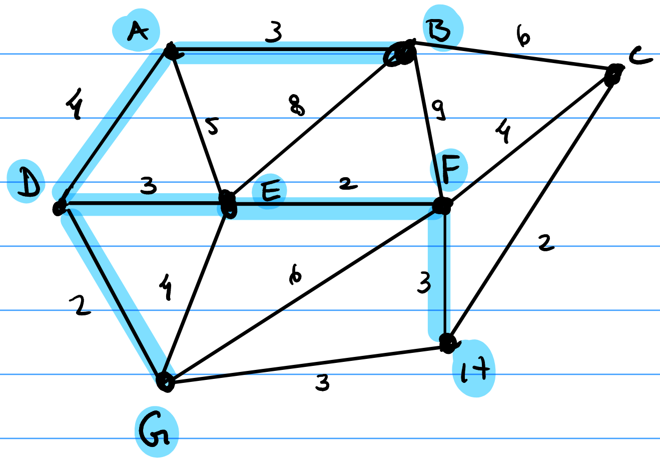
Arestas	Peso
AE	5
BC	6
BF	9
BE	8
DE	3
DG	2



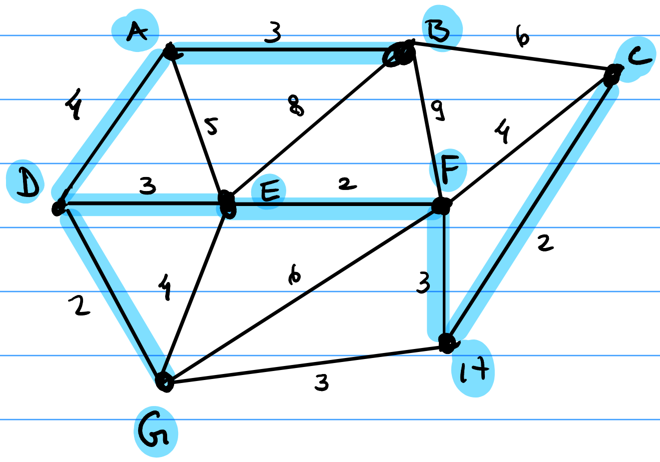
Arestas	Peso
BC	6
BF	9
BE	8
DE	3
GE	4
GF	6
GH	3



Arestas	Peso
BC	6
BF	9
EF	2
GF	6
GH	3



Arestas	Peso
BC	6
FC	4
FH	3
GH	3



Arestas	Peso
BC	6
FC	4
HC	2

Acabou.

5. Seja G um grafo Δ-regular com um ponto de articulação. Mostre que $\chi'(G) = \Delta + 1$.

Suponha, por absurdo, que $\chi'(G) = \Delta$.
Logo, podemos colorir todas as arestas com Δ cores de forma que arestas incidentes em um mesmo vértice tenham cores distintas.
Seja x um ponto de articulação de G. Assim, o subgrafo $G-x$ possui, ao menos, 2 componentes conexas.
Além disso, como $\delta(x) = \Delta$, precisaríamos de Δ cores distintas para colorir as arestas incidentes em x.
Contudo, como $\chi'(G) = \Delta$, então as componentes

conexas admitem uma Δ-coloração por arestas. Assim quando colocamos o vértice x de volta, precisaríamos de, ao menos, mais uma cor para colorir as arestas. Assim G não admite $\chi'(G) = \Delta$.

Pelo Teorema de Vizing: $\Delta \leq \chi'(G) \leq \Delta+1$, então $\chi'(G) = \Delta+1$.

Exercício 1 (1 ponto) Quantas faces existem em um grafo planar regular de grau 3 que tem 10 vértices?

$\Rightarrow$ $3.10 = 2.e$ e $\boxed{e = 15}$. $\boxed{v = 10}$

Pela fórmula de Euler para grafos planares, temos que $v + f = e + 2$

$$f = e + 2 - v = 15 + 2 - 10 = 7.$$

$$\boxed{f = 7}$$

Exercício 2 (2 pontos) O candidato A está qualificado para os trabalhos J_1, J_2, J_4 e J_5. O candidato B está qualificado para os trabalhos J_1, J_4 e J_5. O candidato C está qualificado para os trabalhos J_1, J_4 e J_5. O candidato D está qualificado para os trabalbos J_1 e J_5. O candidato E está qualificado para os trabalhos J_2, J_3 e J_5. O candidato F está qualificado para os trabalhos J_4 e J_5.

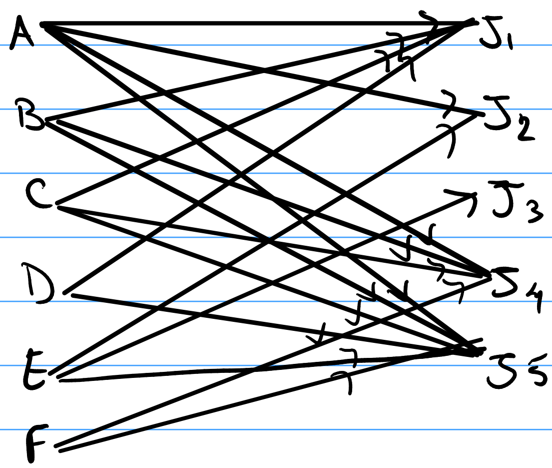
(a) Modele essa situação como uma rede de matchings.

(b) Use o Algoritmo de Ford-Fulkerson para encontrar um matching maximal. Detalhe as iterações do algoritmo: atualização de conjuntos, rotulação, etc.

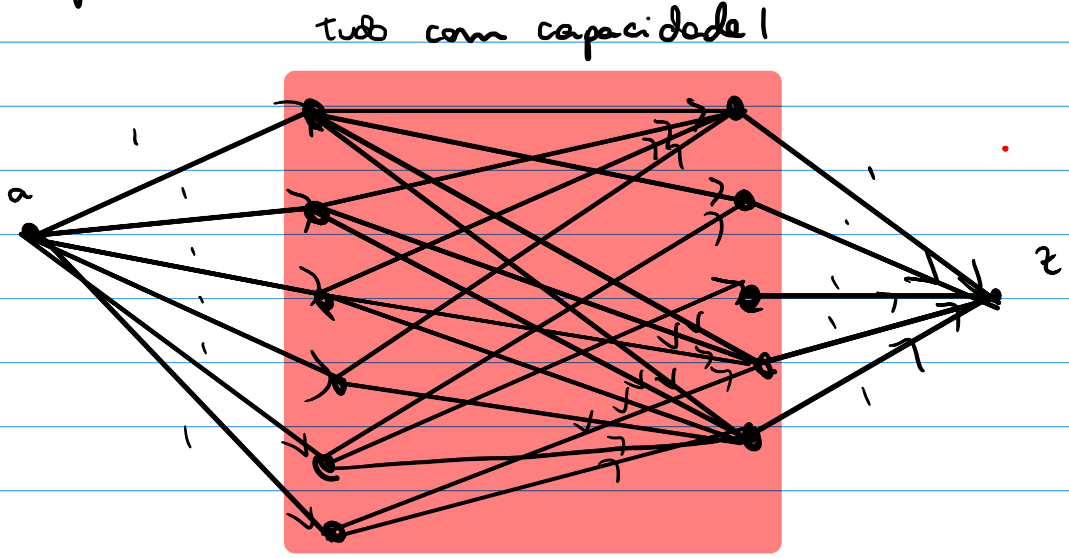
(c) Exiba um corte mínimo.

(d) Existe um matching completo?

a) Os candidatos e os trabalhos são os vértices e uma aresta existe se, e só se, um candidato é qualificado para tal trabalho.

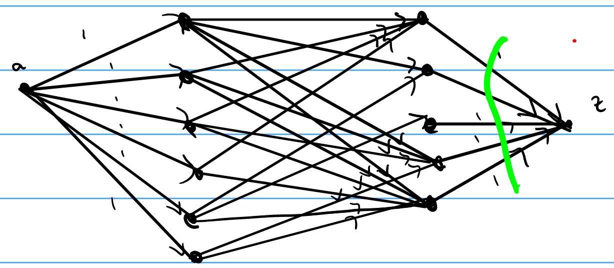


b) Colocando uma superfonte "a" e um supersumidouro "z".



Todos os incrementos são 1.
Basta fazer o algoritmo até esgotar os trabalhos.
Por exemplo: (A, J2); (B, J1); (C, J5); (E, J3); (F, J4).

c) Como o algoritmo de Ford-Fulkerson nos retorna o fluxo máximo (no nosso caso é 5), então o corte mínimo, pelo teorema, será o fluxo máximo que tem valor 5.
O corte mínimo pode ser: a={a, A,B,C,D,E,F,J1,J2,J3, J4,J5}, z={z}.



d) Uma noção intuitiva seria: Não haveria matching completo dos trabalhadores para os trabalhos pois a quantidade de trabalhadores é maior que a de trabalhos. Contudo, existe matching completo para o contrário.
Pelo Teorema de Hall, para termos um marching completo, o conjunto de vizinhos (R(S)) de um subconjunto S de vértices deve ter cardinalidade maior ou igual a cardinalidade de S (|S|) para todo subconjunto S de vértices do grafo.
|R(S)|>=|S| para todo S c= V.
Tomando todos os vértices, temos que |S|=6, mas o conjunto de vizinhos tem cardinalidade 5. Como 5<6, então não há matching completo do conjunto que queremos.

Matching: Subconjunto de arestas sem vértices em comum.
Matching maximal: Matching que não está em outro
Matching máximo: Máximo número possível de arestas
Matching completo: Matching com todos os vértices de V (V → W).
Matching perfeito: Matching com todos os vértices

Exercício 3 (1 ponto)

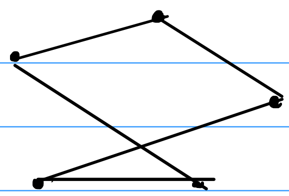
Definição. O *complemento* de um grafo simples G é o grafo simples $\overline{G}$ com os mesmos vértices de G. Um aresta existe em $\overline{G}$ se, e somente se, esta aresta não existe em G.

Um grafo G é *auto-complementar* se G e $\overline{G}$ são isomorfos.

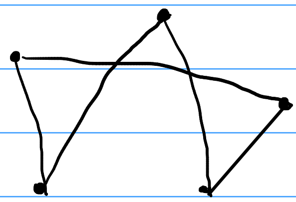
(a) Encontre um grafo auto-complementar tendo 5 vértices.

(b) Encontre outro grafo auto-complementar.

a)

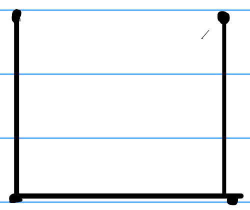


G

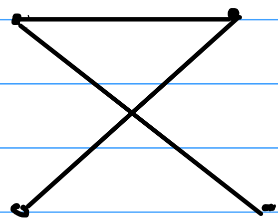


$\overline{G}$

b)



G



$\overline{G}$

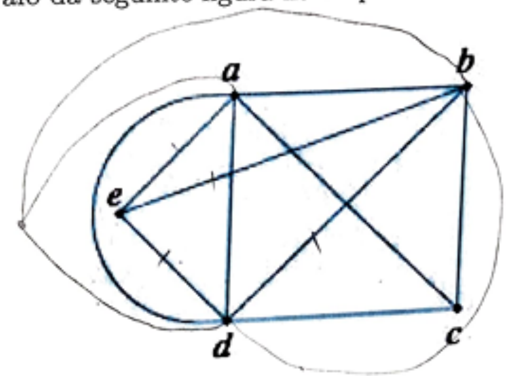
Exercício 4 Nos itens a seguir, decida se a declaração é verdadeira ou falsa. Se a declaração for verdadeira, prove-a. Caso contrário, dê um contraexemplo.

(a) **(0,75 pontos)** Seja G um grafo conexo com pesos. Se todos os pesos em G são distintos, árvores geradoras distintas de G têm pesos diferentes.

(b) **(1,25 pontos)** Seja G um grafo conexo com pesos. Se e é uma aresta de G cujo peso é menor que o peso de qualquer outra aresta, e está em toda árvore geradora minimal de G.

(c) **(1 ponto)** Todo grafo G com $n \geq 2$ vértices e m arestas, com $m < n - 1$, nao e conexo.

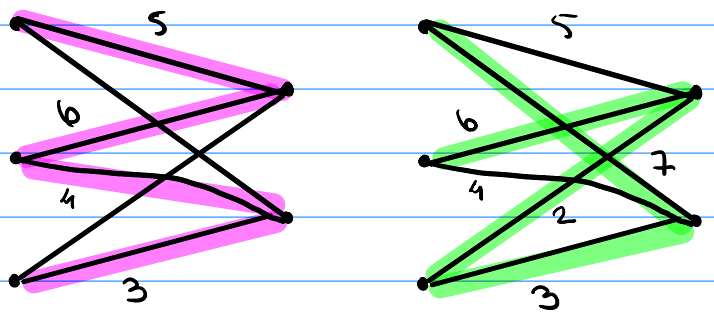
(d) **(1 ponto)** O grafo da seguinte figura não é planar.



a) Falso.

Tome o seguinte grafo com pesos:

$5+6+4+3$
$=11+7=18$



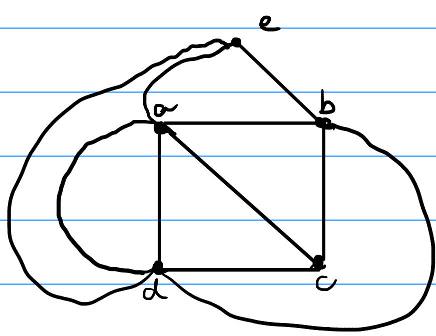
$7+3+2+6$
$=18.$

b) Verdadeiro.

Sabemos que o algoritmo de Prim retorna uma árvore geradora minimal. Assim, em algum momento das iterações do algoritmo, chegaremos num vértice em que tal aresta é incidente. Assim, como devemos escolher o caminho que tenha a menor soma, devemos escolhar tal aresta com peso menor do que todas as outras. Isso vale iniciando em qualquer vértice, já que o algoritmo de Prim é verdadeiro.

c) Verdadeiro.
Vamos provar por indução em n.
Caso base $n=2$: Dois vértices e zero arestas (não é conexo, OK).
Hipótese: Todo o grafo com $n>=2$ vértices e $m<n-1$ arestas não é conexo.
Vamos verificar se vale para um grafo com $n+1$ vértices.
Se o grau de um vértice do grafo for nulo, então o problema está encerrado, já que o grafo sem tal vértice satisfaz a hipótese e, adicionando o vértice, o grafo continua não conexo.
Agora, seja o grau de tal vértice pelo menos um. Ao retirá-lo, o grafo terá n vértices e $m'<n-1$ arestas, já que o grafo original também satisfaz tal condição. Logo, o grafo resultante não é conexo.
Agora, se ao recolocarmos, o grafo se torna conexo, então o grafo possui $m>=n-1$ arestas, o que é um absurdo, por hipótese. Logo o grafo não é conexo.

d) Falso. Segue uma representação planar do grafo:



Exercício 5 (2 pontos)

Definição. A *excentricidade* de um vértice v em uma árvore T é o comprimento máximo de um caminho simples que começa em v.

Um vértice v em uma árvore T é o *centro* de T se a excentricidade de v é minimal.

Mostre que toda árvore tem um ou dois centros.

Ideia: Separar em casos. Quando o número de vértices de T é ímpar, teremos apenas um centro, e quando o número de vértices de T é par, teremos dois coentros.
Assim, toda árvores tem um ou dois centros.
Podemos fazer duas induções (fazer para o caso ímpar e depois adaptar o caso base e fazer para o caso par).

Exercício 1 (1.5 pontos) Prove que um grafo conexo é uma árvore se, e somente se, não contém nenhum laço e tem **exatamente uma** árvore geradora.

(⇒) Árvore ⇒ Não contém laço e tem exatamente uma árvore geradora. (grafo conexo)

Como T é uma árvore, então ela não tem laços. Caso contrário, existiria mais de um caminho entre dois vértices se tal caminho passasse pelo vértice do laço (já que é conexo). → além de T ser grafo simples por def.

Além disso, T possui uma única árvore geradora, uma vez que T é uma árvore, caso contrário, nem todos os vértices seriam contados, ou seriam contados a mais.

(⇐) Não contém laço e tem exatamente uma árvore geradora. ⇒ Árvore (grafo conexo)

Seja G um grafo conexo, sem laços e com exatamente uma árvore geradora.

Logo, para cada par de vértices existe um caminho entre eles. Suponha que G tenha ciclos. Assim, G não possuiria apenas uma árvore geradora. Agora, suponha que G possua arestas paralelas, então, novamente, G não possuiria apenas uma árvore geradora.

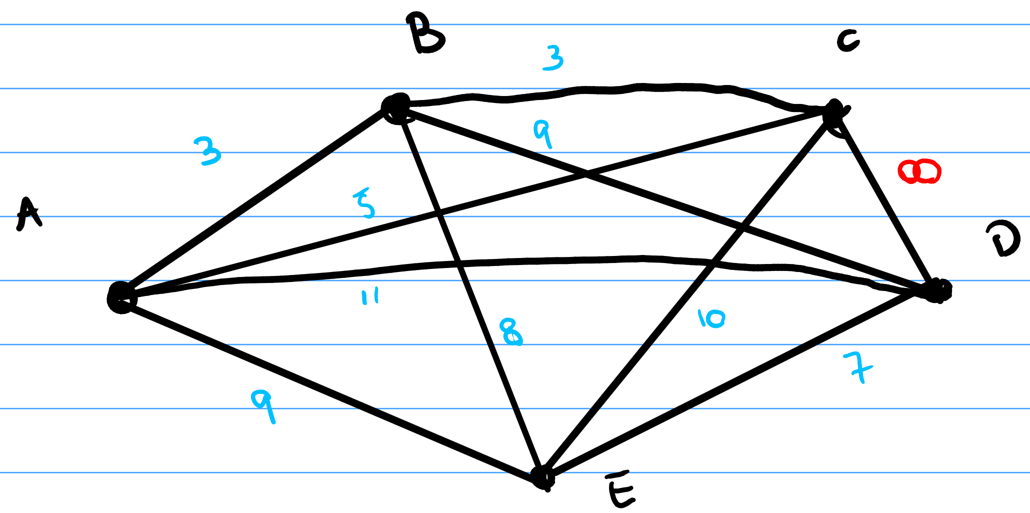
Logo, G é acíclico, simples e conexo, assim, para cada par de vértices existe apenas um caminho. Assim, G é uma árvore.

Exercício 2 (1.5 pontos) Há um projeto que visa construir estradas entre 5 cidades, A, B, C, D e E, de forma que seja possível sair de qualquer uma delas e chegar em qualquer outra. Os custos (em alguma unidade monetária) de construção de estradas entre cada uma das cidades pode ser conferido na tabela abaixo; um custo infinito significa a impossibilidade de construção de uma estrada entre duas cidades.

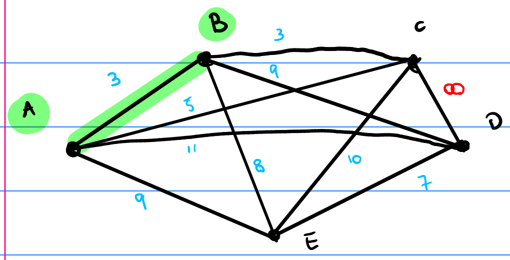
	A	B	C	D	E
A	0	3	5	11	9
B	3	0	3	9	8
C	5	3	0	∞	10
D	11	9	∞	0	7
E	9	8	10	7	0

Se o objetivo é encontrar a maneira mais barata de realizar o projeto, como esse problema pode ser modelado usado teoria de grafos? Qual a melhor solução?

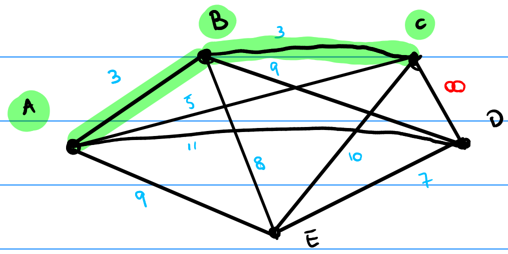
Podemos modelar o problema dessa forma:



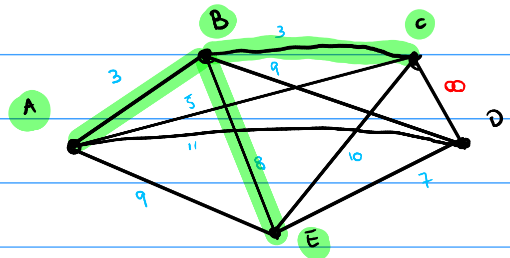
Nosso objetivo é encontrar uma árvore geradora mínima, pois assim encontraríamos pontes que ligam as 5 cidades, seria possível sair de qualquer cidade e chegar a outra e ter custo mínimo.
Para isso, vamos fazer o algoritmo de Prim iniciando em A:



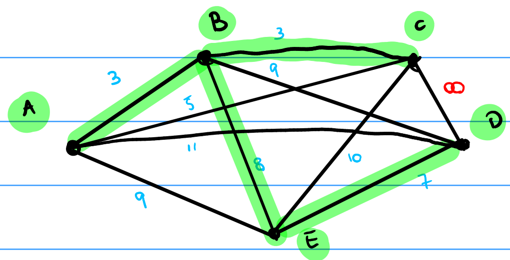
Arestas	Peso
AB	3
AC	5
AD	11
AE	9



Arestas	Peso
AD	11
AE	9
BC	3
BD	9
BE	8



Arestas	Peso
AD	11
AE	9
BD	9
BE	8
CD	∞
CE	10



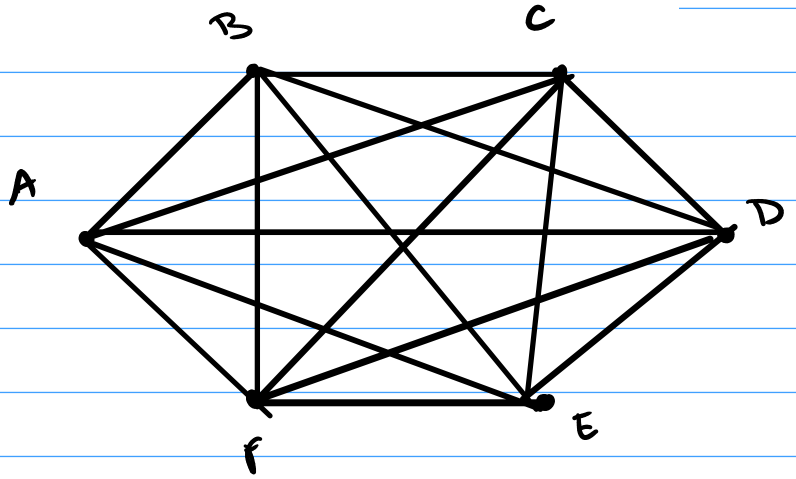
Arestas	Peso
AD	11
BD	9
CD	∞
ED	7

Acabou.

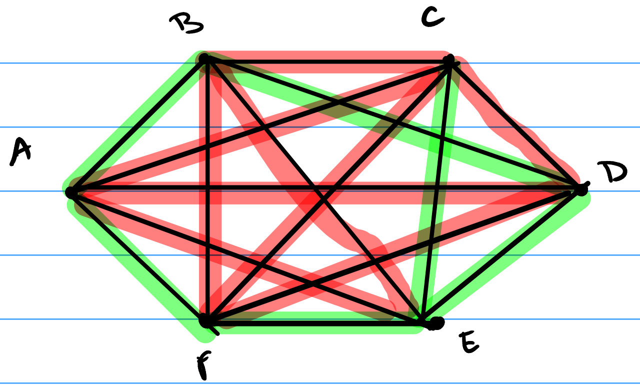
Assim, a MST tem valor 21, com sequência ABCED.

Exercício 3 (2 pontos) Mostre que todo grupo de 6 pessoas contém um trio de pessoas que se conhecem mutuamente ou contém um grupo de pessoas que se desconhecem mutuamente. *Dica: Você pode fazer isso, mostrando que toda coloração do K_6 que usa duas cores contém um triângulo com as três arestas pintadas com a mesma cor.*

Tome o grafo K_6:



Nosso objetivo é provar que dadas duas cores distintas, podemos colorir o K_6 de forma que haja um triângulo monocromático, ou seja, da mesma cor. Um exemplo, se duas pessoas se conhecem, a aresta é pintada de verde, caso contrário, a aresta é pintada de vermelho:



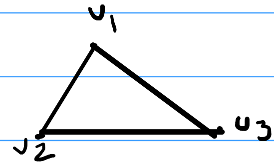
O triângulo CDF é monocromático vermelho, logo CDF não se conhecem mutualmente.

Seja v um vértice de K_6. Então esse vértice é adjacente aos outros 5 vértices. Assim, pelo princípio da Casa dos Pombos, há pelo menos três arestas de uma mesma cor.

Suponha que v_1, v_2, v_3 sejam adjacentes a v. Agora, considere todos os subgrafos que contenham 3 vértices do conjunto $\{v, v_1, v_2, v_3\}$

As possibilidades são: $\{v, u_1, u_2\}$; $\{v, u_2, u_3\}$; $\{v, u_1, u_3\}$; $\{u_1, u_2, u_3\}$.

Agora considere o grafo



se todas as arestas tiverem a mesma cor, o problema acabou. Agora, se alguma aresta tiver cor diferente, algum dos outros subgrafos citados terá todas as arestas da mesma cor.

Exercício 4 (2 pontos) Prove o seguinte resultado:

Teorema: $\chi'(K_n) = \begin{cases} n & \text{se } n \text{ é ímpar} \\ n-1 & \text{se } n \text{ é par} \end{cases}$

Para n ímpar: Vamos representar o grafo como um polígono regular. Assim, pintamos as arestas do polígono (arestas exteriores do grafo) com as n cores. Além disso, as arestas paralelas a uma aresta exterior recebem a mesma cor. Logo, temos uma n-coloração por arestas. No entanto, $\left\lfloor \frac{n}{2} \right\rfloor$ arestas podem ser pintadas com a mesma cor, no mínimo. Como n é ímpar, $\left\lfloor \frac{n}{2} \right\rfloor = \frac{n-1}{2}$. Assim, como K_n tem $\frac{n(n-1)}{2}$ arestas a quantidade mínima de cores para colorir cada aresta é $n(n-1)/2 / ((n-1)/2) = n$ e $\chi'(K_n) = n$ (n ímpar).

Para n par: Como n é par, (n-1) é ímpar e $K_n \supset K_{n-1}$. (K_{n-1} é subgrafo de K_n). Retirando um vértice de K_n, obtemos o K_{n-1}, que pelo item anterior satisfaz $\chi'(K_{n-1}) = n-1$. Contudo, ao acoplar novamente o vértice, observamos que sempre sobra 1 cor diferente para colorir as n-1 arestas incidentes no vértice da (n-1)-coloração por arestas de K_{n-1}. Como esse é o caso mínimo, $\chi'(K_n) = n-1$ (n par).

Exercício 5 Determine e justifique se as seguintes afirmações são verdadeiras ou falsas. São as justificativas que contam pontos.

(a) [1 ponto] Todo grafo com menos arestas do que vértices tem uma componente conexa que é uma árvore.

(b) [0.5 ponto] Todo matching está contido em um matching maximal.

(c) [0.5 ponto] Todo matching máximo é completo.

(d) [1 ponto] Este exercício se refere a uma rede que, além de ter capacidades inteiras não negativas C_{ij}, tem requisitos de fluxo mínimo nas arestas m_{ij}. Ou seja, um fluxo F deve satisfazer

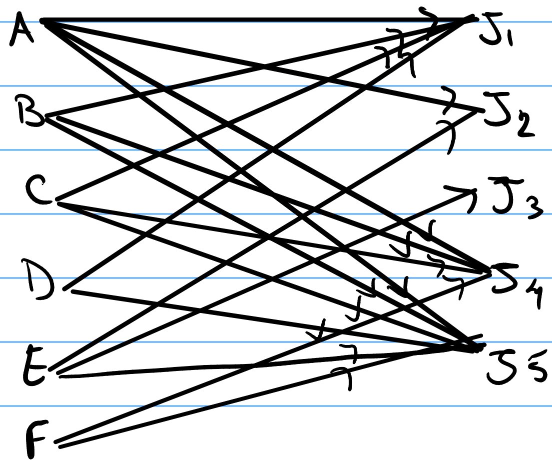
$$m_{ij} \leq F_{ij} \leq C_{ij}$$

para todas as arestas (i, j).

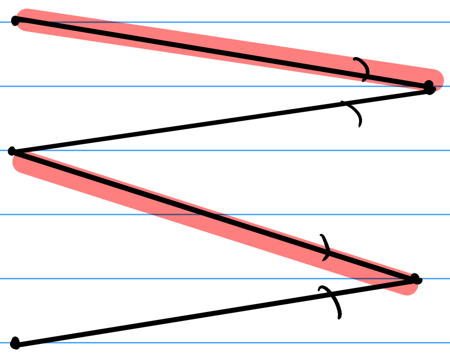
Afirmação: Existe uma rede G na qual $m_{ij} \leq C_{ij}$ para todas as arestas (i, j) e tal que não existe nenhum fluxo.

a) Verdadeiro. Suponha, por absurdo, que não existem componentes conexas com número de arestas inferior ou igual ao número de vértices. Assim, teríamos um absurdo, pois existiriam, no grafo original, mais arestas do que vértices. Logo, existe uma componente conexa com número de arestas inferior ao número de vértices da componente. Como a componente é um grafo conexo e tem menos arestas do que vértices, o número de arestas é o número de vértices -1. Logo a componente é uma árvore.

b) Falso. Um matching maximal é um matching que não está contido em nenhum outro.
Se existirem dois matchings maximais, a afirmação é falsa. Tome o seguinte exemplo:

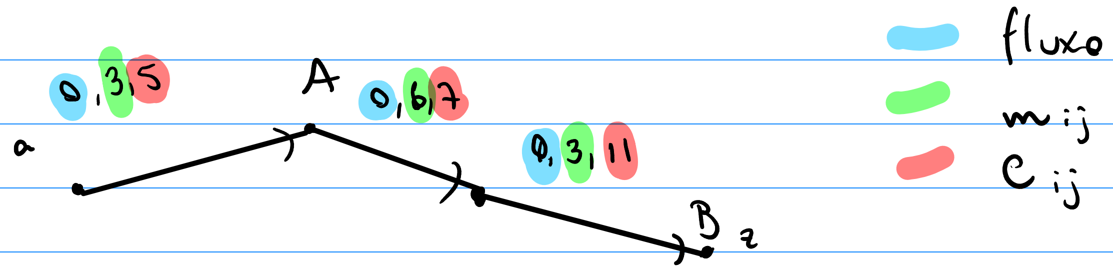


c) Falso. Pela sequência lógica de matchings, temos o seguinte:
Matching completo => Matching máximo =>
=> Matching maximal.
Tome o seguinte exemplo:



é máximo, mas não é completo.

d) Verdadeiro. Como só queremos mostrar que existe, tome o seguinte exemplo:



Não podemos ter fluxo, logo o fluxo é nulo.

Matching: Subconjunto de arestas sem vértices em comum.
Matching maximal: Matching que não está em outro
Matching máximo: Máximo número possível de arestas
Matching completo: Matching com todos os vértices de V (V → W).
Matching perfeito: Matching com todos os vértices