Programa Olímpico de Treinamento
Curso de Combinatória – Nível 3
Prof. Carlos Shine

Aula 18

Teoria dos Grafos: Conectividade

Nessa unidade veremos algumas generalizações e resultados sobre conectividade em grafos.

Generalizando a ideia de conectividade

Às vezes não basta saber se um grafo é conexo ou não; em muitas situações é conveniente que existam alternativas de caminhos (mais ou menos como no trânsito). Definimos então um grafo k-conexo como um grafo G com a seguinte propriedade: G continua conexo com a retirada de menos de k vértices do grafo.

Algo imediato da definição é que todo par de vértices em um grafo k-conexo deve ser conectado por pelo menos k caminhos sem vértices intermediários em comum (caso contrário, excluímos os menos que k vértices de cada caminho e desconectamos os vértices). Ou seja, é necessário que todo par de vértices é conectado por pelo menos k caminhos sem vértices intermediários em comum. Será que é suficiente?

Surpreendentemente, sim.

Teorema de Menger

O *teorema de Menger* prova exatamente isso.

Teorema 1 (Menger)**.** *Um grafo é k-conexo se, e somente se, todo par de vértices é conectado por pelo menos k caminhos sem vértices intermediários em comum.*

A seguinte demonstração mostra o poder do princípio do extremo.

Demonstração: Vamos provar algo um pouco mais forte: sejam u, v vértices distintos e não vizinhos de um grafo G. Definimos $c(u, v)$ como a menor quantidade de vértices diferentes de u e v cuja retirada desconecta u e v e $f(u, v)$ como a maior quantidade de caminhos disjuntos de u a v (no sentido em que não há vértices comum com a exceção de u e v). Provaremos que $c(u, v) = f(u, v)$, o que demonstra o teorema: basta notar que G é $\min c(u, v)$-conexo e a quantidade mínima de caminhos é $\min f(u, v)$.

Vamos à demonstração: primeiro, note que $c(u, v) \geq f(u, v)$, porque todo conjunto de vértices que separa u e v tem que ter pelo menos um vértice de cada caminho que liga u e v.

Agora, provemos que $c(u, v) \leq f(u, v)$. Provaremos por indução sobre k que se $c(u, v) \geq k$ então $f(u, v) \geq k$ e, em particular, para $k = c(u, v)$, $f(u, v) \geq c(u, v)$, o que termina o problema. Note que o que queremos fazer, agora, é mostrar que, dado que precisamos retirar pelo menos k vértices, existem pelo menos k caminhos ligando u a v.

Para $k = 1$, isso é imediato: para separar u e v precisamos tirar pelo menos um vértice, o que quer dizer que existe pelo menos um caminho de u a v.

Vamos ao passo de indução: suponha que $k \geq 1$ e que se $c(u, v) \geq k$ então $f(u, v) \geq k$. Suponha agora que $c(u, v) \geq k+1$, ou seja, que precisamos retirar pelo menos $k+1$ vértices para separar u e v. Pela hipótese de indução, existem k caminhos disjuntos $C_1, C_2, \ldots, C_k$ ligando u a v. Como os k vértices vizinhos a u nesses k caminhos não separam u e v, existe um caminho C ligando u a v cujo vértice vizinho a u é diferente de todos os correspondentes em C_i's. Seja x o primeiro vértice depois de v em C que pertence a algum dos C_i's, e seja C_{k+1} o caminho de u a x. Suponha que escolhemos $C_1, C_2, \ldots, C_{k+1}$ de modo que a distância de x a v em $G - u$ é mínima (veja todas as possibilidades de escolhas de k caminhos para fazer essa escolha). Se $x = v$, conseguimos $k + 1$ caminhos e o passo de indução acaba. Vamos supor, então, que $x \neq v$.

Tire x de G. Pela hipótese de indução, como ainda precisamos tirar k vértices para separar u e v, há pelo menos k caminhos $D_1, D_2, \ldots, D_k$ ligando u a v em $G - x$. Suponha que a escolha dos D_i's foi feita de modo a utilizar a menor quantidade possível de arestas fora dos caminhos C_i. Seja H o grafo obtido unindo os caminhos D_i e o vértice x. Escolha um caminho C_j cuja aresta inicial não está em H. Seja y o primeiro vértice de C_j depois de u que está em H. Se $y = v$, temos $k + 1$ caminhos (os D_i's e C_j) e o problema acabou. Então suponha que $y \neq v$.

Se $y = x$, seja E o menor caminho de x a v em $G - u$. Seja z o primeiro vértice de E que pertence a algum D_i (se não existir z, conseguimos de novo mais um caminho juntando C_j e E). Então a distância de z a v é menor do que de x a v, o que contradiz a escolha dos C_i's. Logo podemos supor que $y \neq x$.

Então y está em algum caminho D_t. Considere o caminho ligando u a y em D_t. Nem todas as suas arestas são arestas de algum C_i, por exemplo, a aresta wy que contém y em D_t, pois y está em C_j e w não, e sabemos que os caminhos C_i são disjuntos aparte de u, v e x. Agora, se trocarmos o trecho u-y de D_t pelo trecho u-y de C_j, obtemos k caminhos disjuntos com menos arestas fora dos C_i's do que os D_i's, contradição. Com isso, o teorema está demonstrado. $\square$

Existe uma variante para arestas desse teorema.

Teorema 2 (Menger para arestas)**.** *Um grafo é dito* k-aresta-conexo *se a retirada de menos de k arestas do grafo o mantém conexo. Então um grafo é k-aresta-conexo se, e somente se, para todo par de vértices u e v, existem k caminhos que ligam u a v sem arestas em comum.*

Demonstração: Considere o *grafo linha de G*, $L(G)$, que é o grafo cujos vértices são as arestas de G e ligamos duas arestas se elas têm um vértice em comum (imagine que o vértice comum vira uma aresta). Aplique o teorema de Menger (o primeiro, é claro!) a esse grafo. $\square$

Você pode se perguntar quanto a resultados mistos. Ou seja, o que acontece se retiramos k vértices e ℓ arestas, obtendo os grafos (k, ℓ)-conexos? Incrivelmente, esse problema ainda está em aberto!

Problema em aberto 1. *Um grafo é (k,ℓ)-conexo se, e somente se, existem $k+\ell$ caminhos sem arestas em comum ligando quaisquer dois vértices, sendo que k deles não têm vértices em comum.*

Caminho mínimo: o algoritmo de Dijkstra

Às vezes estamos interessados em achar o menor caminho entre dois pontos, dentro de uma rede. Por exemplo, a distância entre dois pontos de uma cidade através das ruas. Para formalizar isso em termos de grafos, definimos os pontos intermediários como vértices, as arestas como as ruas e definimos uma função $d\colon A \to \mathbb{R}_+$ que representa a *distância* entre dois vértices. A distância entre dois vértices u e v é o caminho entre u e v cuja soma das distâncias das arestas é mínima. Tal caminho é dito um *caminho mínimo*.

O algoritmo que encontra o caminho mínimo entre duas arestas u e v é o seguinte:

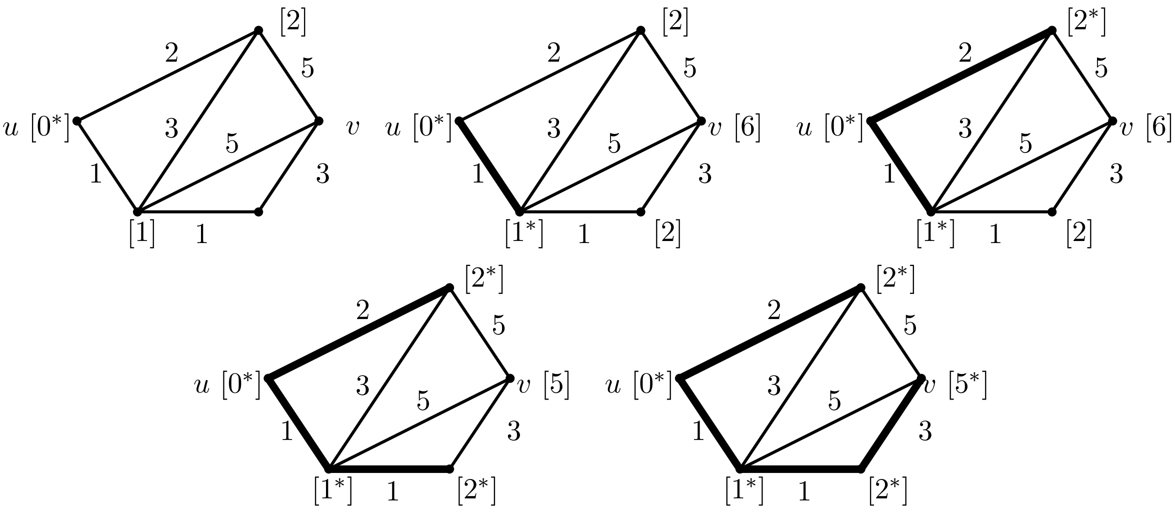
1. Fixe u e associe a cada vértice x um rótulo, que indica a distância de u a x. A ideia é calcular todos os rótulos.

2. Crie um conjunto variável S de vértices cujos rótulos estão definidos. Em princípio, coloque u em S e associe o rótulo 0 a u.

3. Se todos os vértices do grafo estão em S, termine. Caso contrário, considere todos os vizinhos de vértices de S e calcule rótulos temporários para cada um deles. O rótulo de cada vizinho x é igual à menor soma de um rótulo de um vértice vizinho v de S com a distância $d(vx)$.

4. Tome o vértice de menor rótulo temporário e inclua em S, tornando o rótulo permanente. Em caso de empate, escolha qualquer um dos vértices. Volte ao passo 3.

As figuras a seguir mostram a execução do algoritmo.



Vamos mostrar que o algoritmo funciona.

Teorema 3. *O algoritmo de Dijkstra descrito acima funciona.*

Demonstração: Provaremos por indução sobre $|S|$ que o algoritmo calcula a distância entre u e todo outro vértice. Para $|S| = 1$, o resultado é verdadeiro. Agora, suponha que $|S| > 1$ e seja x o último vértice que entrou em S. Por hipótese de indução, o algoritmo calcula a distância entre u e qualquer vértice de $S \setminus \{x\}$. Considere um caminho C de u a x; sejam v o último vértice de S em C antes de chegar a x e t o próximo vértice. Então a distância de u a x é maior ou igual à soma do rótulo $f(v)$ e $d(v,t)$. Mas o algoritmo escolheu x, então, sendo w o vizinho de x em S, $f(v) + d(vt) \geq f(w) + d(wx)$. Logo o caminho até w com a aresta wx é mínimo, e o passo de indução está completo. $\square$

O teorema max flow/min cut

Considere um grafo G e dois vértices s e t. Suponha que as arestas sejam canos, cada um com uma *capacidade*. Queremos saber quanta água podemos passar de s a t. Esse é um típico problema de *fluxo máximo*.

Vamos formalizar tudo isso. Dado um grafo $G = (V, A)$, definimos uma função $c\colon A \to \mathbb{R}_+$, a *capacidade* de cada aresta. Um *fluxo* de s a t é uma função $f\colon A \to \mathbb{R}$ com as seguitnes propriedades:

(F1) $f(x,y) = -f(y,x)$ para $x, y \in V$ (ou seja, se revertermos o fluxo, ele troca de sinal);

(F2) $\sum_{x \in N(v)} f(v,x) = 0$ para todo $v \in V \setminus \{s,t\}$. Isso quer dizer que nos vértices intermediários não "entra nem vaza" água;

(F3) $|f(e,v)| \leq c(e)$ para todo $e \in A$ (não ultrapassamos a capacidade).

O *valor* do fluxo é igual à quantidade de água que sai de s e chega em t, ou seja, é $|f| = \sum_{x \in N(s)} f(s,x) = \sum_{x \in N(t)} f(x,t)$.

Enfim, definimos um *corte* como uma partição $(S, \overline{S})$ do conjunto V de vértices, e a *capacidade do corte* $c(S, \overline{S})$ como a soma das capacidades *positivas* de todas as arestas saindo de S.

Note que se $u \in S$ e $v \in \overline{S}$, todo fluxo é menor ou igual a $c(S, \overline{S})$. O teorema max flow/min cut mostra que a igualdade pode ser alcançada: o fluxo máximo é igual ao menor corte.

Teorema 4 (Max flow/min cut)**.** *Se as capacidades são todas inteiras, o fluxo máximo de s a t é igual ao corte mínimo com s de um lado e t do outro.*

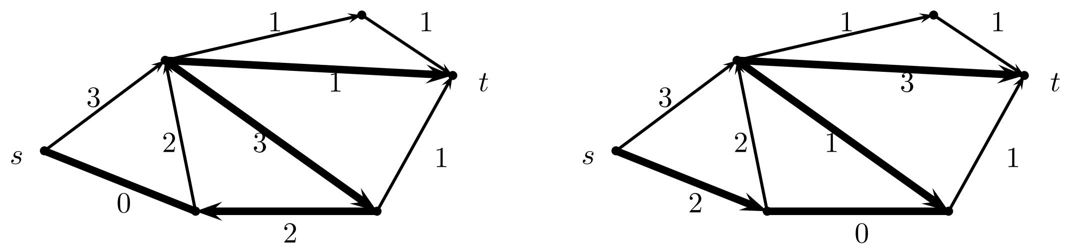
Demonstração: Iremos encontrar um fluxo que é igual a algum corte. A ideia é construir uma sequência de fluxos, até obter um igual a um corte.

Comece com o fluxo nulo ($f_1(u,v) = 0$ para toda aresta). Para cada fluxo f_k, seja S o conjunto dos vértices x tais que existe um caminho $v_0 v_1 \ldots v_n$ de $s = v_0$ a $x = v_n$ tal que $f(v_{i-1}, v_i) < c(v_{i-1} v_i)$ para todo i, $1 \leq i \leq n$.

Se $t \in S$, podemos aumentar o fluxo: considere o caminho $v_0 v_1 \ldots v_n$ com $s = v_0$ e $t = v_n$ e seja $\epsilon = \min\{c(v_{i-1} v_i) - f(v_{i-1}, v_i)\}$. Construa o novo fluxo

$$f_{k+1}(u,v) = \begin{cases} f_k(u,v) + \epsilon & \text{se } uv \text{ está no caminho} \\ f_k(u,v) & \text{se } uv \text{ não está no caminho} \end{cases}$$

Intuitivamente, estamos aumentando o fluxo em ϵ.



Um caminho; todas as capacidades são iguais a 3, e $\epsilon = 2$.

Temos $|f_{k+1}| \geq |f_k| + 1$ pois ϵ é inteiro. Como o fluxo está limitado pelas capacidades, em algum momento não poderemos mais aumentar o fluxo, ou seja, em algum momento $v \notin S$. Mas aí, da definição de S, $f(v,x) = c(v,x)$ para todo $v \in S$ e $x \in \overline{S}$ com $f(v,x) > 0$, e aí $|f| = c(S, \overline{S})$. $\square$

O teorema max flow/min cut tem várias aplicações:

Teorema 5 (Casamentos)**.** *Dado um grafo bipartido com classes V_1 e V_2, para $S \subset V_1$ seja $N(S)$ o conjunto de todos os vértices vizinhos a algum elemento de S. Um* empa-relhamento *de V_1 em V_2 é um conjunto de arestas disjuntas cujas extremidades estão em classes diferentes. Então existe um emparelhamento completo de V_1 em V_2 se, e somente se, $|N(S)| \geq |S|$ para todo $S \subset V_1$.*

Demonstração: Se $|N(S)| < |S|$ é imediato que não é possível ter um emparelhamento completo (não há vizinhos suficientes para S!). Provaremos a recíproca usando max flow/min cut.

Crie dois vértices, s ligado a todos os vértices de V_1 e t ligado a todos os vértices de V_2, e coloque capacidades 1 em cada uma dessas novas arestas e ∞ nas arestas originais (ou seja, pode passar qualquer coisa lá!).

Seja $(S, \overline{S})$ um corte mínimo. Sabemos que $c(S, \overline{S}) \leq |V_1|$, pois existe o corte $(\{s\}, V \setminus \{s\})$. Provaremos que $c(S, \overline{S})$ é na verdade igual a $|V_1|$.

Seja $X = S \cap V_1$. Como a capacidade de todas as arestas originais é infinita, nenhum corte pode ter essas arestas separando S e $\overline{S}$. Logo $N(X) \subset S \cap V_2$. Então

$$c(S, \overline{S}) = \sum_{x \in S, y \in \overline{S}} c(x,y) = \sum_{y \in \overline{S} \cap V_1} c(s,y) + \sum_{x \in S \cap V_2} c(x,t) \geq |V_1| - |X| + |N(X)| \geq |V_1|$$

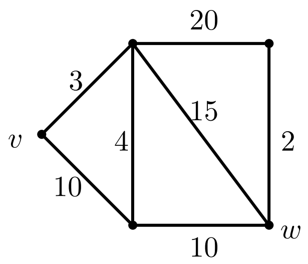
Portanto todo corte mínimo tem capacidade $|V_1|$, e o fluxo correspondente nos dá o emparelhamento máximo. $\square$

Outra demonstração do teorema de Menger: Primeiro, provamos a versão com arestas: escolha s e t como dois vértices quaisquer e considere cada aresta com capacidade 1. Seja f um fluxo máximo e $(S, \overline{S})$ o corte mínimo correspondente. O fluxo determina a quantidade de caminhos sem arestas em comum, sendo que uma aresta está no caminho quando o fluxo nela é 1. O corte determina a menor quantidade de arestas que separam s e t. Então o resultado é imediato de max flow/min cut.

Quanto à versão com vértices, divida cada vértice v diferente de s e t em dois vértices v_- e v_+ (cada um dos dois vértices está ligado aos correspondentes dos vizinhos de v) e crie uma aresta de capacidade 1 de v_- para v_+. Mantenha cada aresta original com capacidade 1. Isso garante que no fluxo não haja vértices em comum nos caminhos. O corte determina as arestas que separam s e t; basta tomar um vértice de cada aresta e juntar. $\square$

Problemas

1. Determine o menor caminho de A a B na rede a seguir.



2. Seja G um grafo 2-conexo e v_1 e v_2 dois vértices de G. Sendo $n = |V| = n_1 + n_2$, prove que existe uma partição (V_1, V_2) de V tal que $v_i \in V_i$, $|V_i| = n_i$ e os subgrafos induzidos por V_i são conexos, $i = 1, 2$.

3. Prove que o teorema de Menger é equivalente ao seguinte teorema:

   Teorema 6 (Menger para conjuntos de vértices)**.** *Sejam V e W conjuntos de vértices do grafo G. Para cada inteiro positivo k, existem k caminhos ligando vértices de V a vértices de W se, e somente se, todo conjunto de vértices que separa V e W contém pelo menos k vértices.*

4. Seja G um grafo k-conexo, $k \geq 2$, e seja T um conjunto com no máximo k vértices de G. Prove que existe um ciclo em G contendo todos os vértices de T.

5. Uma matriz com entradas reais tem a seguinte propriedade: as somas dos elementos de cada linha e de cada coluna são todas inteiras. Prove que podemos trocar cada elemento da matriz pelo seu piso ou teto de modo que as somas dos elementos em cada linha e em cada coluna não mudam.

6. (Putnam) Um torneio tem $2n$ times e durou $2n - 1$ dias. Em cada dia, todo time jogou contra outro time; em cada jogo um time venceu e outro perdeu. No fim do torneio, cada time jogou com cada um dos demais exatamente uma vez. É sempre possível escolher um time vencedor de cada dia sem escolher times repetidos?

7. Sejam $A_1, A_2, \ldots, A_{20}$ conjuntos disjuntos, cada um com 20 elementos do conjunto $\{1, 2, \ldots, 400\}$ e $B_1, B_2, \ldots, B_{20}$ conjuntos nas mesmas condições. Prove que é possível formar 20 pares de conjuntos (A_i, B_{j_i}) com $\{j_1, j_2, \ldots, j_{20}\} = \{1, 2, \ldots, 20\}$ e $A_i \cap B_{j_i} \neq \emptyset$ para todo i, $1 \leq i \leq 20$.

8. Sejam a, n inteiros positivos com $a \geq (n-1)!$. Prove que existem n primos distintos $p_1, p_2, \ldots, p_n$ tais que $p_i \mid a + i$, $1 \leq i \leq n$.

Bibliografia

1. B. Bollobás, *Graph Theory: An Introductory Course.*

2. R. Diestel, *Graph Theory.* Springer, 2003.

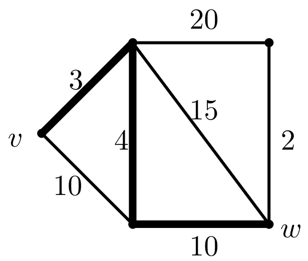
3. O. R. Oellermann. *Menger's Theorem.* Disponível em
   `http://ion.uwinnipeg.ca/~ooellerm/MengersTheoremMay11-2010.pdf`

4. F. Kammer, H. Täubig. *Graph Connectivity.* Disponível em
   `www.informatik.uni-augsburg.de/thi/personen/kammer/Graph_Connectivity.pdf`

5. T. Andreescu e Z. Feng, *102 Combinatorial Problems, From the training of the USA IMO team*, Birkhäuser 2003.

Respostas, Dicas e Soluções

1. O caminho mínimo está indicado a seguir, a distância total é 17.



2. Indução sobre n_1. Para $n_1 = 1$, sendo o grafo 2-conexo, temos que $G - v_1$ é conexo. Agora suponha que conseguimos uma partição nas condições do enunciado com $|V_1| = n_1$ e $|V_2| = n_2$. Vamos passar um vértice w de V_2 para V_1 sem alterar conectividade. Escolhemos w o vértice mais distante de v_2 (em termos de quantidade de arestas no caminho mínimo) que tem vizinho em V_1. Obviamente $V_1 \cup \{w\}$ é conexo. Provemos que $V_2 \setminus \{w\}$ também é. Suponha que exista um vértice v de V_2 desconectado de v_2. Como G é 2-conexo, existe um caminho de v_2 a v em G que não contém w. Seja x o último vértice desse caminho, indo no sentido de v a v_2, que está em V_2. Pela definição de w, o caminho de v_2 a x em G_2 não contém w. Mas juntando esse último caminho com o primeiro obtemos um caminho de v_2 a v que não contém w, absurdo, e o problema acabou.

3. (Menger $\Longrightarrow$ Menger para conjuntos de vértices) Dados V e W, crie dois novos vértices v e w; ligue v a todos os elementos de V e w a todos os elementos de W. Então cada caminho de V a W corresponde a um caminho de v a w.

   (Menger para conjuntos de vértices $\Longrightarrow$ Menger) Dados v e w, seja $V = N(v)$ e $W = N(w)$. Então cada caminho de v a w corresponde a um caminho de V a W.

4. Considere o menor ciclo que contém a maior quantidade possível de vértices de T. Suponha que ele não contém todos os vértices de T. Tome um dos vértices $v \in T$ que ficaram de fora do ciclo e W como o ciclo e considere os k caminhos disjuntos que ligam v a W (que existem pelo teorema de Menger). Sejam $v_1, v_2, \ldots, v_m$ os vértices de T no ciclo, nessa ordem, e considere os trechos $v_i v_{i+1}$ do ciclo (sendo $v_m v_{m+1} = v_m v_1$). Como $m < k$, dois caminhos C e D que ligam v a W terminam no mesmo trecho $v_i v_{i+1}$ (por casa dos pombos). Então basta trocar o trecho $v_i v_{i+1}$ por $v_i C v D v_{i+1}$ no ciclo e obtemos um ciclo com mais vértices de T, contradição. Logo existe um ciclo que passa por todos os vértices de T.

5. Primeiro, podemos supor sem perda de generalidade que todas as entradas da matriz estão entre 0 e 1 (basta subtrair o piso de cada entrada). Considere o seguinte grafo: os vértices são s, t, cada linha e cada coluna; as arestas são de s para cada linha, de cada linha para cada coluna e de cada coluna para t. Aplicaremos o max flow/min cut da seguinte forma: as capacidades de s para cada linha é a soma das suas entradas; as capacidades das linhas para as colunas são iguais a 1; as capacidades de cada coluna para t é a soma das suas entradas.

   Considere o seguinte fluxo: de s para cada linha é a soma dos elementos; da linha i para a coluna j é a_{ij}; de cada coluna para t é a soma dos elementos. Agora, considere o corte em que $S = V \setminus \{t\}$ e $\overline{S} = \{t\}$. Note que $c(S, \overline{S})$ é igual à soma dos elementos de toda a matriz, que é igual ao fluxo! Então esse fluxo é máximo e esse corte é mínimo, e sabemos que existe um fluxo máximo inteiro. Esse fluxo indica quais entradas da nova matriz devem ser iguais a 1 e quais devem ser iguais a 0.

6. Considere o grafo bipartido em que os times ficam em uma classe T de $2n$ elementos e os dias na outra D de $2n-1$ elementos. Ligamos o time ao dia se ele venceu nesse dia. Queremos verificar se existe um casamento de D para T. Seja S um conjunto de dias e $N(S)$ o conjunto dos times que ganharam em algum desses dias. Suponha que $|N(S)| < |S|$. Como há mais times que dias, existe um time t que não está em $N(S)$. Mas então t perdeu de pelo menos $|S|$ times nesses dias, porque t jogou em todos esses dias; logo $|N(S)| \geq |S|$ (pelo menos os times que ganharam de t), absurdo. Logo, pelo teorema dos casamentos, é sempre possível escolher $2n-1$ times que ganharam em dias diferentes.

7. Considere o grafo bipartido em que os vértices de uma classe A são os conjuntos A_i e os vértices da outra classe B são os conjuntos B_j. Ligamos A_i a B_j se $A_i \cap B_j \neq \emptyset$. Seja S um subconjunto de A e $N(S)$ seus vizinhos. Os elementos de $N(S)$ são os conjuntos B_j que contêm algum elemento da união dos A_i's de S. Como essa união tem $|S| \cdot 20$ elementos, são necessários pelo menos $|S|$ conjuntos B_j para cobri-los. Logo $|N(S)| \geq |S|$ e, pelo teorema dos casamentos, podemos formar os pares (A_i, B_{j_i}).

8. Construa o grafo bipartido em que uma classe N tem como vértices os números $a+i$ e a outra classe P tem como vértices os primos que dividem algum dos $a+i$. Ligue $a+i$ a p_k se $p_k \mid a+i$. Vamos provar que, para todo subconjunto S de N, $|N(S)| \geq |S|$. Isso termina o problema, porque aí existe um casamento e conseguimos $p_i \mid a+i$.

Seja então $S = \{a+i_1, a+i_2, \ldots, a+i_k\}$ e suponha que $m < k$ fatores primos $p_1, p_2, \ldots, p_m$ apareçam neles. Considere, para cada primo p_i, o inteiro que contém a maior quantidade de fatores p_i. Como há mais números que primos, um deles não foi escolhido. Suponha, sem perda, que seja $a+i_k$. Então $a+i_k \mid (a+i_1)(a+i_2)\ldots(a+i_{k-1})$, o que equivale a

$$\begin{aligned} & (a+i_1)(a+i_2)\ldots(a+i_{k-1}) \equiv 0 \pmod{a+i_k} \\ \iff & (i_k-i_1)(i_k-i_2)\ldots(i_k-i_{k-1}) \equiv 0 \pmod{a+i_k} \\ \implies & |(i_k-i_1)(i_k-i_2)\ldots(i_k-i_{k-1})| \geq a+i_k \end{aligned}$$

Mas

$$|(i_k-i_1)(i_k-i_2)\ldots(i_k-i_{k-1})| \leq (n-1)! < a+i_k,$$

absurdo, e o problema acabou.