

CPS740 Algoritmos e Grafos
COS242 Teoria de Grafos
Prova P1 – Gabarito

Professores: F. Marquezino, C. Figueiredo
Universidade Federal do Rio de Janeiro

11 de Agosto de 2020

Critérios de correção da prova P1 realizada de 11 a 17 de Agosto de 2020.

1. [3 pontos] Diga se cada afirmação abaixo é verdadeira ou falsa e dê uma justificativa. (Cada resposta correta e com justificativa correta soma 0.5 ponto; cada resposta incorreta ou com justificativa incorreta desconta 0.3 ponto; itens deixados em branco não somam nem descontam pontos.) **[Durante a correção eu considere as respostas com justificativas incompletas como se estivessem em branco, ou seja, não descontei ponto nesses casos. Além disso, em alguns casos, somei um pouco menos de 0.5 ponto quando havia alguma imprecisão na linguagem porém sem comprometer o argumento central.]**

- (1.1) F Se G é 3-colorível, então G possui um ciclo ímpar.

Solução: Um grafo 3-colorível é aquele que pode ser colorido com 3 cores ou menos. Qualquer grafo bipartido pode ser colorido com 2 cores, portanto é 3-colorível, e ao mesmo tempo é livre de ciclos ímpares. O objetivo da questão era verificar a fluência no uso das definições.

- (1.2) F Seja $G(V_1 \cup V_2, E)$ um grafo bipartido conexo. Então o grafo complementar G^c também é conexo.

Solução: A solução mais simples é exibir um contraexemplo. Pode-se exibir um grafo G bipartido conexo e seu complemento G^c , de modo que fique claro que G^c não é conexo.

Solução: De modo mais geral, pode-se demonstrar que se G for um grafo bipartido completo, então seu complemento não será conexo.

- (1.3) V Se retirarmos uma aresta qualquer do grafo $K_{3,3}$, o grafo resultante é planar.

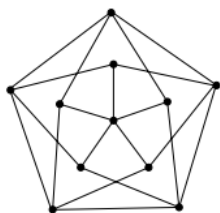
Solução: Pode-se exibir uma representação planar do grafo obtido pela remoção de uma de $K_{3,3}$ e argumentar que não há perda de generalidade na escolha da aresta removida (pois ao remover qualquer outra aresta obteríamos um grafo isomorfo ao grafo planar exibido).

Solução: Pode-se argumentar por que o grafo resultante não contém como subgrafo uma subdivisão de K_5 ou de $K_{3,3}$.

- (1.4) V Seja f um isomorfismo de um grafo G para um grafo H , e seja w um vértice em G . O grau de w em G é igual ao grau de $f(w)$ em H .

Solução: Seja $\{w_1, \dots, w_k\}$ o conjunto de vértices adjacentes a w . Então, pela definição de isomorfismo, temos que $\{f(w_1), \dots, f(w_k)\}$ é o conjunto de vértices adjacentes a $f(w)$. Portanto, o grau do vértice w e o grau do vértice $f(w)$ são ambos iguais a k .

- (1.5) F O grafo abaixo é planar:



Solução: O grafo possui $n = 11, m = 20$. Se o grafo fosse planar, então pela fórmula de Euler $f = 2 - n + m = 11$. Como o grafo é livre de triângulos, cada face deveria ter pelo menos 4 arestas. Como cada aresta pertence a duas faces, teríamos $4f \leq 2m$, contradição.

Solução: Pode-se exibir a subdivisão de K_5 ou de $K_{3,3}$ contida no grafo.

- (1.6) **F** Todo hipercubo de dimensão $n, n \geq 1$, possui ciclo Hamiltoniano. Nota: um hipercubo de dimensão n é um grafo $G(V, E)$ em que $V = \{0, 1, \dots, 2^n - 1\}$, e em que $(v, w) \in E$ se e somente se as representações binárias dos vértices v e w diferem em exatamente um bit.

Solução: Para $n = 1$ o grafo possui somente uma aresta, portanto não possui ciclo.

Solução: Se a afirmação fosse “Todo hipercubo de dimensão $n, n > 1$, possui ciclo Hamiltoniano”, então a resposta seria **V**. A justificativa seria: Verifique que é verdade para hipercubo de dimensão $n = 2$ (que possui somente uma aresta). Assuma que seja verdade para $n = k$. Para mostrar que é verdade para $n = k + 1$, construa um hipercubo H de dimensão $k + 1$ a partir de duas cópias H_1, H_2 do hipercubo de dimensão k , adicionando arestas entre os vértices correspondentes. Tome um ciclo Hamiltoniano $v_1 v_2 \dots v_k$ em H_1 e o reverso $v'_k \dots v'_2 v'_1$ em H_2 . Então o ciclo $v_1 \dots v_{k-1} v_k v'_k v'_{k-1} \dots v'_1$ será um ciclo Hamiltoniano de H . Outra justificativa aceitável nesse caso seria apresentar um algoritmo para percorrer todos os vértices do hipercubo de dimensão n sem repetir os vértices, e então demonstrar sua corretude. Para isso pode-se utilizar, por exemplo, um código de Gray. Como a intenção não era fazer “pegadinha”, irei analisar caso a caso e talvez aceitar respostas com essas justificativas.

2. [$3\frac{1}{2}$ pontos] Sejam $G_1(V_1, E_1)$ e $G_2(V_2, E_2)$ dois grafos conexos e seja $G(V, E)$ o grafo obtido a partir de G_1 e G_2 a partir da identificação de um vértice qualquer em V_1 com um vértice qualquer em V_2 . Prove que $\chi(G) = \max\{\chi(G_1), \chi(G_2)\}$, ou apresente um contraexemplo.

Solução: Seja $m = \max\{\chi(G_1), \chi(G_2)\}$. Então, por um lado temos $\chi(G) \geq m$. Por outro lado, sejam A, B duas colorações com m cores (não necessariamente ótimas) para G_1, G_2 respectivamente. Defina uma coloração C tal que o vértice v receba a cor $A(v)$ caso $v \in V_1$, e a cor $B(v)$ caso $v \in V_2$. Como não existe aresta entre G_1 e G_2 , temos que C é uma coloração para G , então $\chi(G) \leq m$. Portanto, $\chi(G) = m$.

3. [$3\frac{1}{2}$ pontos] Escreva um algoritmo que, dados N pontos no plano, encontre o par de pontos mais próximos. Seu algoritmo deve ter complexidade de tempo $O(N(\log N)^2)$. Utilize uma das técnicas estudadas nas aulas. Se necessário, podem utilizar o seguinte resultado sem demonstrar: Um retângulo de comprimento d e altura $2d$ pode conter no máximo seis pontos tais que quaisquer dois pontos estejam a distância pelo menos d entre si.

Solução: Pode-se utilizar a técnica de dividir e conquistar.

Entrada: uma lista $P[1 \dots N]$ de N pontos.

Saída: o par de pontos mais próximos em P e a distância entre eles.

- 1: ordene P por ordem crescente das coordenadas x
- 2: particione a lista P em duas sublistas $P_L[1 \dots \lfloor N/2 \rfloor]$ e $P_R[\lfloor N/2 \rfloor + 1 \dots n]$

- 3: resolva o problema recursivamente para P_L e P_R ; de modo que retorne $p_a^{(L)}, p_b^{(L)}$ como os pontos mais próximos entre si na sublista P_L , sendo d_L distância entre eles, e analogamente para $p_a^{(R)}, p_b^{(R)}, d_R$
- 4: seja $d := \min(d_L, d_R)$
- 5: salve em uma lista M todos os pontos de P cuja coordenada x esteja a distância menor que d em relação à coordenada x do ponto $P[\lfloor N/2 \rfloor]$
- 6: ordene M por ordem crescente das coordenadas y
- 7: encontre os pontos mais próximos em M e a distância entre eles; seja o resultado, respectivamente, $p_a^{(M)}, p_b^{(M)}$ e d_M
- 8: seja $d' := \min(d, d_M)$ e sejam p'_a, p'_b os pontos que estão a distância d' entre si
- 9: retorne como resultado p'_a, p'_b, d'

Para receber pontuação completa, deve-se argumentar por que o algoritmo é correto e por que tem complexidade $O(N(\log N)^2)$.

O algoritmo é correto pelo seguinte argumento: (i) os passos 1–4 identificam os pontos mais próximos em cada uma das metades do plano delimitadas pela linha horizontal que passa pelo ponto $P[\lfloor n/2 \rfloor]$, sendo d a distância entre eles; (ii) os passos 5–7 identificam pontos que porventura estejam a distância menor que d entre si, sendo que estes necessariamente devem existir dentro da caixa de largura $2d$ delimitada por margens de largura d à esquerda e à direita da linha horizontal considerada nos passos anteriores.

A complexidade de tempo $T(N)$ do algoritmo acima pode ser obtida da seguinte forma: Passo 1 pode ser feito em tempo $O(N \log N)$ utilizando, por exemplo, o algoritmo Mergesort estudado em aula; passo 2, em tempo $O(N)$; passo 3, em tempo $2T(\lfloor N/2 \rfloor)$; passo 4, em tempo constante; passo 5, em tempo $O(N)$; passo 6, em tempo $O(N \log N)$; passo 7, em tempo $O(N)$, pois cada ponto só precisa ser comparado com até outros seis; passos 8 e 9, em tempo constante. Portanto, a complexidade de tempo do algoritmo é $T(N) = 2T(\lfloor N/2 \rfloor) + O(N \log N) = O(N(\log N)^2)$.

Alternativamente, o passo 6 poderia ser melhorado para $O(N)$, e nesse caso a complexidade seria $T(N) = 2T(\lfloor N/2 \rfloor) + O(N) = O(N \log N)$, porém isso não é exigido na questão.