UFRJ – PESC-COPPE
COMPLEXIDADE DE ALGORITMOS
PROFESSORES: Franklin Marquezino, Fábio Botler, e Celina de Figueiredo
SEGUNDA PROVA P3/2021

Data de publicação: 11/02/2022

DATA DE ENTREGA: 18/02/2022

Questão 1. (2 pontos) Seja (G, c) uma instância do TSP e seja T uma árvore geradora mínima para (G, c). Prove que $c(T)$ é um limite inferior para $opt(G, c)$. T é uma solução viável para (G, c)? Justifique a sua resposta.

Questão 2. (2 pontos) Considere o problema Vertex Cover e a heurística que seleciona a cada iteração o vértice corrente de grau máximo. Dê um exemplo de uma família de grafos bipartidos para a qual a heurística retorna um conjunto de vértices C tal que a razão $|C|/|C^*|$ não é limitada por uma constante, isto é, a razão cresce, a medida que o número de vértices do grafo bipartido da família cresce.

Questão 3. (2 pontos) Um grafo split é um grafo que admite uma partição do seu conjunto de vértices em dois conjuntos disjuntos C e S na qual C induz uma clique (os vértices são mutuamente adjacentes), e S induz um conjunto independente (os vértices são mutuamente não adjacentes).

É muito conhecido que um grafo G é split se e somente se G não contém, como subgrafo induzido, um grafo pertencente ao conjunto $\{G_1, G_2, G_3\}$, em que G_1 é um ciclo sem cordas e com quatro vértices (conhecido como C_4), G_2 é um ciclo sem cordas e com cinco vértices (conhecido como C_5), G_3 é um emparelhamento com duas arestas (conhecido como $2K_2$). Essa caracterização fornece imediatamente um algoritmo polinomial de complexidade $O(n^5)$ que decide se um grafo G é split.

Considerando que você tem esse algoritmo polinomial e usando árvore de busca limitada, descreva um algoritmo FPT para resolver a seguinte tarefa: Dado um grafo $G = (V, E)$ e um inteiro k; verificar se G possui um conjunto F com no máximo k vértices cuja remoção o torne um grafo split.

Dica: considere a remoção recursiva de vértices, de modo a destruir um subgrafo induzido de cada nó da árvore de busca que não seja um grafo split.

QUESTÃO 4. (4 pontos) Um *caminho* é um grafo conexo de grau máximo no máximo 2 e possui dois vértices com grau exatamente 1. Um *ciclo* é um grafo conexo no qual cada vértice possui grau exatamente 2. Um caminho ou ciclo em um grafo G é dito *Hamiltoniano* se contém todos os vértices de G. Considere os seguintes problemas, e resolva os itens $a - g$.

Problema: `Caminho Hamiltoniano`.
Dados: um grafo G
Pergunta: G possui um caminho Hamiltoniano?

Problema: `Ciclo Hamiltoniano`.
Dados: um grafo G
Pergunta: G possui um ciclo Hamiltoniano?

Problema: `Ciclo meio Hamiltoniano`.
Dados: um grafo G
Pergunta: G possui um caminho com pelo menos $|V(G)|/2$ vértices?

a) Apresente uma redução do problema `Caminho Hamiltoniano` para o problema `Ciclo Hamiltoniano`.

b) Suponha que exista um algoritmo A' que decide `Caminho Hamiltoniano`, e construa um algoritmo A que decide `Ciclo Hamiltoniano` usando apenas um número pequeno de execuções de A';

c) Enuncie as definições de Problemas NP-Completos e NP-Difícil;

d) Mostre que `Ciclo Hamiltoniano` está em NP.

   No que segue, assuma que `Caminho Hamiltoniano` é um problema NP-Completo.

e) Mostre que `Ciclo Hamiltoniano` é NP-difícil;

f) `Ciclo Hamiltoniano` é NP-Completo? justifique

g) `Ciclo meio Hamiltoniano` é NP-Completo? justifique