UNIVERSIDADE FEDERAL DO CEARÁ - CAMPUS CRATEÚS
CIÊNCIA DA COMPUTAÇÃO

Aluno(a): ______________________________
CRT0044 - Teoria da Computação

Matrícula: ______________
Período: 2023.1
Prof. Rennan Dantas

Nota: ______

3ª. ETAPA

1. Para cada uma das afirmações abaixo, diga se ela é verdadeira, falsa, verdadeira se $P \neq NP$ ou falsa se $P \neq NP$. Dê uma justificativa para cada resposta.
   a) Não há problemas em P que são NP-Completos.
   b) Existem problemas em P que estão em NP.
   c) Se A pode ser polinomialmente reduzido a B, e B eNP-Completo, então A e NP-Completo.
   d) Se A pode ser polinomialmente reduzido a B, e $B \in P$, então $A \in P$.

2. Tome um certo problema de otimização (maximização) $P(I)$, onde I representa uma instância de P e $P(I)$ tem como resposta o custo máximo de uma solução para I, ou uma indicação de que I é ilimitada, ou uma indicação de que I é inviável. Tome agora $P(I, k)$, que consiste em decidir se a instância I tem uma solução de custo pelo menos k. Argumente que a existência de um algoritmo polinomial para $P(I)$ implica na existência de um algoritmo polinomial para $P(I, k)$. Argumente que, para instâncias I que não são ilimitadas, a volta é verdadeira.

3. Decidir se um grafo tem um ciclo hamiltoniano (que passa por todos os vértices) é um problema que se encontra em **NP**. Assumindo **P** $\neq$ **NP**, você consegue imaginar um problema de decisão que:

   a) Não se encontra em NP? Argumente.

   b) Se encontra em NP$\setminus$P? Argumente.

   Obs: em todas as questões, a justificativa é o mais importante. Mas nessa questão, o impacto da justificativa na pontuação é ainda maior.

4. Mostre que se $P = NP$, existe um algoritmo de tempo polinomial que recebe um grafo direcionado G, uma fonte s e um destino t como entrada e encontra um caminho hamiltoniano de s a t contido no grafo (suponha que só exista um caminho). Perceba que o algoritmo deve retornar o conjunto de arestas que compõe o caminho hamiltoniano, se houver caminho hamiltoniano. **Mostre** o algoritmo.

5. Tome $L_1, L_2 \in$ **P**. Prove que $L_1 \cup L_2 \in$ **P** e que $L_1 \cap L_2 \in$ **P**. Faça o mesmo para **NP**.

6. Dizemos que um grafo G está parcialmente rotulado se alguns de seus vértices possuem um número inteiro como rótulo. Dado um vértice rotulado v de G, seja $r(v)$ o seu rótulo. Seja CAMPO-MINADO o problema de decidir se, dado como entrada um grafo G parcialmente rotulado, G pode ser completamente rotulado de forma que qualquer vértice v com rótulo positivo tenha exatamente $r(v)$ vizinhos com rótulo negativo. **Prove que CAMPO-MINADO é NP-Completo.**

Dica: Imagine rótulos negativos como bombas do campo minado. Tente reduzir 3-SAT para este problema: cada variável tendo dois vértices no grafo com um vizinho comum com rótulo igual a 1. Pense nas bombas do campo minado como uma atribuição de verdadeiro.

7. Se $P = NP$, os problemas em NP-Completo podem ser resolvidos em tempo polinomial. Suponha que você tem uma subrotina "caixa-preta" que resolve o problema de decisão da questão anterior. Construa um algoritmo para encontrar o conjunto independente máximo de um grafo não-direcionado dado como entrada. O tempo de execução do **seu algoritmo deve ser polinomial em** $|V|$ **e** $|E|$.

8. O problema do SET-PARTITION recebe como entrada um conjunto X de números. A questão é se os números podem ser particionados em dois conjuntos A e $\overline{A}$=$X - A$ tal que

$$\sum_{x \in A} x = \sum_{x \in \overline{A}} x$$

Mostre que SET-PARTITION é NP-Completo.

Dica 1: Utilize o SUBSET-SUM. O SUBSET-SUM é definido como a seguir: Dado um conjunto X de inteiros e um alvo t, existe um subconjunto $Y \subseteq X$ tal que os membros de Y somados resultam em t?

Dica 2: Seja s a soma dos membros de X. Faça $X' = X \cup \{s - 2t\}$ como entrada do SET-PARTITION.

Dica 3: Note que se o SET-PARTITION retorna verdadeiro, existem duas partições de X' cuja soma dos valores dos elementos é $s - t$.

9. Dado um grafo $G = (V, E)$, vamos falar de dois problemas de cobertura:

   - Cobertura de vértices por vértices: decidir se há $V' \subseteq V, |V'| \leq k$, tal que todo $v \in V$ ou está em V' ou tem um vizinho em V'. Dica: redução a partir do 3-SAT. Engrenagem de variáveis para escolher apenas uma das variáveis como verdadeira e engrenagem de cláusula para que cada cláusula tenha pelo menos um vértice verdadeiro.
   - Cobertura de arestas por vértices: decidir se há $V' \subseteq V, |V'| \leq k$, tal que toda aresta tem um extremo em V'.

   Prove que esses problemas são **NP**-completos.