Universidade Federal do Ceará
Departamento de Computação

Complexidade Computacional

Prof. Rudini Sampaio

Lista de exercícios

A data de entrega de cada exercício está marcada ao lado de cada questão. Cada questão receberá uma nota entre 0 e 1. Questões entregues até no máximo 2 dias após o prazo receberão no máximo 0.75. Até no máximo 1 semana, receberão no máximo 0.5. Não serão aceitas questões entregues após 1 semana do prazo. Enviar os exercícios até 23h59 para os emails **jmbmaluno@alu.ufc.br** e **rudini@ufc.br** com o título "*Complexidade Computacional - Questão X*", onde X é o número da questão. As questões devem ser escritas à mão e devem ser enviadas como imagem (foto) ou PDF digitalizado.

1. [**09-Out**] Uma **linguagem** é um conjunto de palavras, onde uma palavra é uma sequência de símbolos (letras ou caracteres) de um alfabeto. Um **problema de decisão** faz uma pergunta do tipo sim ou não para uma instância dada. Explique porque a classificação em classes de complexidade computacional (como P, NP ou PSPACE) feita para linguagens também serve para problemas de decisão.

2. [**09-Out**]

(a) O que é a Tese de Church-Turing?
(b) O que é um problema de decisão Reconhecível (ou Recursivamente Enumerável)?
(c) O que é um problema de decisão Decidível?
(d) O que é o Problema da Parada?
(e) Dê um exemplo de um problema Indecidível.
(f) Dê um exemplo de um problema não Reconhecível.

3. [**10-Out**] Responda e explique cada um dos itens abaixo.

(a) O que é a Classe P?
(b) O que é um certificado de um problema de decisão? O que é a Classe NP e qual a relação dela com certificados?
(c) O que é um Algoritmo Não-Determinístico?
(d) O que é uma Redução Polinomial entre dois problemas e para que serve?

4. [**13-Out**] O que é a Classe NP-Completa? Explique os dois modos de se provar que um problema de decisão é NP-Completo. Qual primeiro problema provado NP-Completo. Quem provou e quando?

5. [**13-Out**] Dê uma ideia geral em no máximo 10 linhas de como é a demonstração do Teorema de Cook-Levin.

6. [**15-Out**] Para cada uma das afirmações abaixo, diga se ela e *verdadeira*, *falsa*, *verdadeira se P≠NP* ou *falsa se P≠NP*. Dê uma justificativa curta para cada resposta.

(1) Não há problemas em P que são NP-Completos
(2) Existe apenas algoritmo exponencial para o problema da parada
(3) Existem problemas em P que estão em NP
(4) Existem problemas em NP que não estão em P
(5) Se A pode ser polinomialmente reduzido a B, e B é NP-Completo, então A é NP-Completo
(6) Se A pode ser polinomialmente reduzido a B, e B ∈ P, então A ∈ P
(7) O problema de obter o percurso mínimo do Caixeiro Viajante é NP-Completo
(8) O problema SAT não pertence a Classe P

7. [**16-Out**] Seja MOCHILA o problema de decidir se, dados inteiros positivos P e V e dado um conjunto S onde cada elemento $s \in S$ possui um peso $p(s)$ e um valor $v(s)$, existe um subconjunto S' de S tal que a soma dos pesos dos elementos de S' seja menor ou igual a P e a soma dos valores dos elementos de S' seja maior ou igual a V. Prove que MOCHILA é NP-Completo. **Dica:** Problema SOMA-SUBC (ou SUBSET-SUM) da soma de subconjuntos.

8. [**18-Out**] Seja HITTING-SET o problema de decidir se, dado como entrada um inteiro K e uma coleção de subconjuntos $C_1, \ldots, C_m$ de um conjunto S, existe um subconjunto S^* com K elementos de S tal que, para todo subconjunto C_i, C_i contém algum elemento de S^*. Prove que HITTING-SET é NP-Completo. **Dica:** Cobertura de vértices.

9. [**20-Out**] Dizemos que um grafo G esta 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 exatamete $r(v)$ vizinhos com rótulo negativo. Prove que CAMPO-MINADO é NP-Completo. Dica: Imagine rótulos negativos como bombas do campo minado. Tente reduzir 3SAT 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.

10. [**22-Out**] No seguinte jogo de paciência, é dado um tabuleiro $n \times n$. Em cada uma das suas n^2 posições está colocada uma pedra azul ou uma pedra vermelha ou nenhuma pedra. Você joga removendo pedras do tabuleiro até que cada coluna contenha pedras de uma única cor e cada linha contenha pelo menos uma pedra. Você vence se atingir esse objetivo. Vencer pode ser possível ou não, dependendo da configuração inicial. Seja PACIÊNCIA o problema de decidir se dado um tabuleiro dessa forma como entrada é possível vencer. Prove que PACIÊNCIA é NP-Completo. Dica: Tente reduzir 3SAT para este problema: pense as colunas como variáveis e as linhas como cláusulas. Depois adicione algumas linhas e colunas inóquas para que o tabuleiro fique quadrado.

11. [**24-Out**] Seja DOMINANTE o problema de decidir se, dado como entrada um grafo G e um inteiro $k > 0$, existe um conjunto D com k vértices de G tal que todo vértice de G está em D ou é adjacente a algum vértice de D.

- Prove que DOMINANTE é NP-Completo usando o problema 3SAT
- Prove que DOMINANTE é NP-Completo usando o problema COB-VERT da Cobertura de vértices.

12. [**27-Out**] Seja COR-DIF o problema de decidir se, dado como entrada um conjunto S e uma coleção $C = \{C_1, \ldots, C_k\}$ de subconjuntos de S, onde $k > 0$, é possivel colorir os elementos de S com duas cores de forma que nenhum conjunto C_i tenha todos os seus elementos com a mesma cor. Prove que COR-DIF é NP-Completo. **Dica:** Tente reduzir 3SAT para este problema: Crie um conjunto S contendo toda variável e seu complemento. Adicione um elemento especial F a S. Para cada variável, crie um subconjunto C_i contendo apenas ela e seu complemento. Para cada cláusula, crie um subconjunto C_j contendo seus literais e mais o elemento especial F.

13. [**30-Out**] Dado um grafo G, uma coloração é uma atribuição de cores a seus vértices de forma que vértices adjacentes tenham cores diferentes. Seja 3CORES o problema de decidir se, dado um grafo G como entrada, G pode ser colorido com 3 cores. Mostre que 3CORES é NP-Completo.

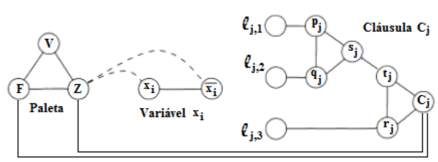

FIGURA 1. Use essas engrenagens na questão 1

14. [**17-Nov**] Seja GEOGRAFIA o jogo em que, dado um grafo direcionado G e um vértice inicial v_0, Alice e Bob alternadamente selecionam em cada turno um vértice sempre formando um caminho maior a partir de v_0. Na variante normal, perde quem não puder jogar. Na variante misère, vence quem não puder jogar.

(a) Prove que GEOGRAFIA é PSPACE-completo na variante normal (**Dica:** explique a redução mostrada em sala de aula a partir de QSAT).
(b) Prove que GEOGRAFIA é PSPACE-completo na variante misère (**Dica**: redução de GEOGRAFIA-normal, incluindo um vértice para o qual todos os outros apontam; justifique).

15. [**27-Nov**] Considere o Jogo de Coloração Conexa visto em sala de aula. Prove que é PSPACE-completo a partir de uma redução do Jogo de Coloração em que Bob é o primeiro a jogar.

16. [**01-Dez**] Considere os problemas de otimização do conjunto Dominante mínimo em Grafos e do Set-Cover mínimo. Mostre reduções que preservam aproximação de Dominante para Set-Cover e de Set-Cover para Dominante.

17. [**07-Dez**] Considere os problemas de otimização do conjunto Independente máximo e do Set Packing máximo. Mostre reduções que preservam aproximação de Independente para Set Packing e de Set Packing para Independente.

18. [**10-Dez**] Com relação às classes de complexidade probabilística:

(a) Prove que $\mathrm{P} \subseteq \mathrm{ZPP}$.
(c) Prove que $\mathrm{BPP} \subseteq \mathrm{PSPACE}$.
(b) Prove que $\mathrm{RP} \cap \mathrm{coRP} = \mathrm{ZPP}$.
(d) Prove que $\mathrm{RP} \cup \mathrm{coRP} \subseteq \mathrm{BPP}$.
(e) Prove que $\mathrm{RP} \cup \mathrm{coRP} \subseteq \mathrm{NP}$.
(f) Prove que $\mathrm{RP} = \mathrm{PCP}_{1\%,0}$ (**Dica:** repita até prob. de erro cair para 1/2)

19. [**25-Dez: Feliz Natal**] Considere o Jogo de Ativação definido a seguir. Seja G um grafo em que todo vértice está *ativado* ou *desativado*. A ativação de um vértice desativado v é um processo em dois passos: no primeiro passo, v é ativado e todos os seus vizinhos são desativados e, no segundo passo, todo vértice desativado sem nenhum vizinho ativado se torna ativado. É um jogo síncrono: todos os vértices atualizam seu status ao mesmo tempo. Note que a ativação de v afeta apenas os vértices à distância 1 e 2 de v. No Jogo da Ativação, a instância é um grafo G e um inteiro k. Os vértices de G são rotulados com A, B ou C. Além disso, os vértices de G já vem com uma configuração inicial: todo vértice está ativado ou desativado. Alice e Bob alternadamente ativam um vértice desativado de modo que Alice (resp. Bob) pode ativar apenas vértices rotulados com A (resp. B). Se o número de vértices ativados é no máximo k em algum momento do jogo, Alice vence. Se alguma configuração se repete, Bob vence. Mostre que o Jogo da Ativação é EXPTIME-completo (Dica: redução do Jogo ASAT (Alternating SAT), também chamado de ABF (Alternating Boolean Formula), visto em sala de aula).