



Instituto Federal de Educação, Ciência e Tecnologia de Brasília – Câmpus Taguatinga
Ciência da Computação – Análise de Algoritmos
Prova II – 1º/2019 – Divisão e Conquista, Algoritmos Gulosos e Programação Dinâmica
Prof. Daniel Saad Nogueira Nunes

Aluno: _____

Matrícula: _____

Data: 29 de maio de 2019

Duração da prova: 120 minutos

Tabela de notas (uso exclusivo do professor)

Questão	Pontos	Nota
1	3	
2	3	
3	3	
4	3	
Total	12	

Observações

- Esta prova tem o total de 3 páginas (incluindo a capa) e 4 questões.
- O número total de pontos é 12.
- Certifique-se de assinar todas as folhas de resposta bem como a capa da prova.
- Leia atentamente todas as questões da prova. A interpretação do problema é crucial para o desenvolvimento correto da resposta.
- Resoluções sem justificativa não serão consideradas.
- É vedado o uso de equipamentos eletrônicos, como celulares, notebooks entre outros.
- A prova será **anulada** e medidas disciplinares serão tomadas para os alunos que “colarem” durante a avaliação.

★ Certifique-se de assinar todas as folhas de resposta.

Questão 1 (3 pontos)

Uma sequência X de parênteses contendo apenas os símbolos ‘(’ e ‘)’ é dita balanceada se satisfaz uma das seguintes condições:

- X é a string vazia.
- $X = (Y)$, para uma sequência balanceada de parênteses Y .
- $X = YZ$, para sequências balanceadas de parênteses Y e Z .

Utilizando esta definição, podemos dizer por exemplo que a string $X = (((()))((()))((()))())$ é uma sequência balanceada de parênteses.

Dada uma string X contendo apenas os símbolos ‘(’ e ‘)’:

- (2 pontos) Elabore um algoritmo $O(n)$ que receba a string X e responda se ela é uma sequência balanceada de parênteses ou não.
- (1 ponto) Analise o seu algoritmo.

Questão 2 (3 pontos)

Epaminondas possui vários livros de espessuras variáveis. Ele comprou uma prateleira de comprimento W para colocar alguns de seus livros. Contudo, Epaminondas deseja utilizar o comprimento inteiro da prateleira, isto é, a soma das espessuras dos livros escolhidos para serem colocados na prateleira deve ser exatamente igual a W . Assim, dada a espessura dos livros e o comprimento W da prateleira, crie um algoritmo que diga se existe alguma escolha de livros que atenda a propriedade descrita anteriormente.

- Entrada: $v[]$, um vetor de n inteiros contendo a espessura de cada livro e $W \in \mathbb{N}$, o comprimento da prateleira.
 - Saída: **Sim**, se existe uma escolha de livros cuja soma das espessuras é exatamente igual a W e **Não** caso contrário.
- (1 ponto) Se a espessura dos livros é $v = (1, 5, 21, 8, 4, 2, 1)$ e $W = 15$, liste todas as possibilidades de escolha de livros.
 - (1 ponto) Modele sua solução através de uma relação de recorrência.
 - (1 ponto) Elabore um algoritmo que leve tempo $O(nW)$ que diga se existe uma escolha de livros cuja soma das espessuras é exatamente igual a W .

Questão 3 (3 pontos)

De acordo com o problema da maior subsequência comum (LCS), dadas duas strings X e Y , de comprimento n e m respectivamente:

- (1 ponto) Defina a solução da LCS entre X e Y recursivamente através de uma relação de recorrência.
- (1 ponto) Preencha a matriz de programação dinâmica da LCS para $X = GACTAT$ e $Y = AGACAT$.
- (1 ponto) Escreva em português como recuperar a maior subsequência comum para a entrada do item anterior através da tabela de programação dinâmica e destaque ao menos uma subsequência comum mais longa.

★ Certifique-se de assinar todas as folhas de resposta.

Questão 4 (3 pontos)

Dado um inteiro n , calcule $\sqrt[3]{n}$. Caso n não possua uma raiz cúbica exata, deverá ser calculado o valor $\lfloor \sqrt[3]{n} \rfloor$. Seu algoritmo deverá levar tempo $o(n)$. **Restrição:** apenas operações aritméticas simples como soma e multiplicação são permitidas, obviamente você não poderá utilizar a operação que tira a raiz cúbica de um número.

- (a) (2 pontos) Projete o algoritmo de acordo com a restrição dada.
- (b) (1 ponto) Analise o seu algoritmo.