



CEFET-MG

CENTRO FEDERAL DE
EDUCAÇÃO TECNOLÓGICA
DE MINAS GERAIS



CONCURSO PÚBLICO 2023

DOCENTE EBTT CAMPUS DIVINÓPOLIS

EDITAL Nº 28/2023

ETAPA DA PROVA ESCRITA

ENGENHARIA DE COMPUTAÇÃO



Número de inscrição do Candidato

ABRA SOMENTE QUANDO AUTORIZADO

QUESTÃO 01 (10,0 pontos)

MOSTRE, por indução, que se n for um número inteiro positivo, então:

$$1 + 2 + \dots + n = \frac{(n(n+1))}{2}$$

QUESTÃO 02 (10,0 pontos)

Considere a recorrência $T(n) = 2T(n-1) + 1$ com a condição inicial $T(0) = 1$.

- DEDUZA uma fórmula geral para a recorrência a partir dos quatro primeiros valores de $T(n)$.
- DEDUZA uma fórmula geral para a recorrência do enunciado iterando na recorrência com os valores $n-1$, $n-2$, $n-3$ e $n-4$.

QUESTÃO 03 (10,0 pontos)

Considere o seguinte pseudocódigo do algoritmo de Dijkstra:

```
DIJKSTRA( $G, w, s$ )
1  INITIALIZE-SINGLE-SOURCE( $G, s$ )
2   $S = \emptyset$ 
3   $Q = V[G]$ 
4  while  $Q \neq \emptyset$ 
5       $u = \text{EXTRACT-MIN}(Q)$ 
6       $S = S \cup \{u\}$ 
7      for cada vértice  $v \in G.Adj[u]$ 
8          RELAX( $u, v, w$ )
```

Fonte: CORMEN, T. H.; Algoritmos: Teoria e Prática. 3. Ed. Rio de Janeiro: LTC, 2012.

- EXPLIQUE como o algoritmo de Dijkstra lida com arestas com pesos negativos em um grafo?
- DEFINA grafos ponderados e dirigidos?
- No algoritmo de Dijkstra padrão a fila de prioridades é implementada utilizando-se arranjos. CITE uma alternativa mais eficiente de implementação dessa fila de prioridades, supondo que o algoritmo de Dijkstra será aplicado em um grafo esparso? COMPARE a complexidade da alternativa apresentada com a implementação padrão.
- EXPLIQUE como o algoritmo de Dijkstra se comporta em grafos densos.

QUESTÃO 04 (10,0 pontos)

Para cada afirmativa abaixo, MOSTRE se a igualdade é verdadeira ou falsa para todo n inteiro positivo:

a. $2^{n+1} = O(2^n)$

b. $2^{2n} = O(2^n)$

Para cada uma das afirmativas, utilize a definição formal de notação assintótica para mostrar se é verdadeira ou falsa. Caso a afirmativa seja verdadeira, APRESENTE uma prova simples. Caso seja falsa, APRESENTE um contraexemplo.

QUESTÃO 05 (10,0 pontos)

Em relação aos conceitos de "l-values" e "r-values":

a) DIFERENCIE esses conceitos

b) EXPLIQUE como esses conceitos são utilizados em operações de atribuição.

QUESTÃO 06 (10,0 pontos)

No contexto de linguagens de programação, responda os itens a seguir:

a) DEFINA o conceito de "avaliação em curto-circuito". Apresente um exemplo.

b) Dado o trecho de pseudocódigo abaixo, EXPLIQUE como o uso da avaliação em curto-circuito da expressão lógica que controla a execução do laço pode evitar um erro de acesso a um índice inexistente da lista caso o valor desejado não esteja presente na lista. Considere que os elementos na lista são indexados a partir de 0 (zero) e as expressões são avaliadas da esquerda para a direita.

```
1. posicao = 0;
2. ENQUANTO ( (posicao < tamanho_lista) E (lista[posicao] != valor) ) {
3. posicao = posicao + 1;
4. }
```

QUESTÃO 07 (10,0 pontos)

No contexto de linguagens formais e autômatos, **INDIQUE** quais formalismos são utilizados para especificar e para reconhecer as cadeias das seguintes classes de linguagens:

- a) Linguagens regulares
- b) Linguagens livres de contexto
- c) Linguagens recursivamente enumeráveis

QUESTÃO 08 (10,0 pontos)

MOSTRE uma gramática livre de contexto para as seguintes linguagens sobre o alfabeto $\{0,1\}^*$:

- a) $L(G) = \{w \in \{0,1\}^* \mid w \text{ tem o número igual de 0s e 1s}\}$.
- b) $L(G) = \{w \in \{0,1\}^* \mid w = w^R\}$, ou seja, os palíndromos sobre $\{0,1\}^*$

QUESTÃO 09 (10,0 pontos)

Os autômatos finitos determinísticos têm aplicação em Compiladores, no reconhecimento dos tokens. Considere uma linguagem de programação hipotética que possui um *token* denominado *number*, que corresponde a constantes numéricas. O *token number* é especificado conforme a definição regular a seguir:

digit $\rightarrow [0-9]$

digits $\rightarrow \text{digit}^+$

optionalFraction $\rightarrow \text{.digits} \mid \lambda$

optionalExponent $\rightarrow (E (+|-| \lambda) \text{digits}) \mid \lambda$

number $\rightarrow \text{digits optionalFraction optionalExponent}$

MOSTRE um autômato finito determinístico (AFD) que reconheça o *token number* conforme descrito acima.

Fonte: AHO, A. V.; LAM, M. S.; SETHI, R.; ULLMAN, J. D. Compiladores: Princípios, técnicas e ferramentas. 2. ed. São Paulo: Pearson Addison-Wesley, 2008. p. 85.

QUESTÃO 10 (10,0 pontos)

Um compilador é estruturado em duas partes denominadas *front-end* (ou análise) e *back-end* (ou síntese). Para cada uma dessas partes, **CITE** e **EXPLIQUE** o papel das fases da compilação que as constituem.