

Prova 02 de Introdução ao Pensamento Algorítmico	SIMULADO
Nome:	
RA:	
<p>Questão 01) Qual é uma possível desvantagem da estratégia de divisão e conquista?</p> <p>a) Pode levar a um consumo excessivo de memória devido à recursão.</p> <p>b) É difícil de implementar para a maioria dos problemas.</p> <p>c) É sempre mais rápido que qualquer outra abordagem.</p> <p>d) Não é aplicável a problemas de grande escala.</p> <p>e) Só funciona para operações matemáticas de divisão.</p>	
<p>02) Em que situação o Quicksort apresenta seu melhor caso?</p> <p>a) Quando o pivô escolhido sempre divide o array em subarrays de tamanho similares.</p> <p>b) Quando o pivô é sempre escolhido de forma aleatória.</p> <p>c) Quando há muitos elementos duplicados no array.</p> <p>d) Quando o tamanho do array é pequeno.</p> <p>e) Quando o tamanho do array é grande.</p>	
<p>03) Qual situação provavelmente causaria a degradação do desempenho de uma tabela hash?</p> <p>a) Quando a função de hash distribui as chaves uniformemente.</p> <p>b) Quando a tabela é muito grande.</p> <p>c) Quando o fator de carga é muito baixo.</p> <p>d) Quando há muitas colisões e a estratégia de resolução é ineficiente.</p> <p>e) Quando o tamanho da tabela hash é um número primo.</p>	
<p>04) Em um grafo não ponderado, se a Busca em Largura (BFS) é utilizada para encontrar o caminho de um nó de origem para todos os outros nós alcançáveis, qual propriedade fundamental dos caminhos encontrados ela garante?</p> <p>a) Os caminhos encontrados são sempre os mais longos possíveis.</p> <p>b) Os caminhos encontrados têm o menor número de arestas.</p> <p>c) Os caminhos encontrados são únicos.</p> <p>d) Os caminhos encontrados visitam todos os nós do grafo antes de qualquer aresta ser revisitada.</p> <p>e) Os caminhos encontrados são duplos.</p>	
<p>05) O que representa o valor armazenado em cada nó durante a execução do algoritmo de Dijkstra?</p> <p>a) A quantidade de vizinhos.</p> <p>b) O número de ciclos encontrados.</p> <p>c) A menor distância do nó inicial até aquele nó.</p> <p>d) A profundidade da árvore gerada.</p> <p>e) A complexidade do grafo.</p>	
<p>06) Considere o grafo com os seguintes vértices: <b>X, Y, Z, W, T</b>. E as arestas com pesos: <math>X \rightarrow Y</math> (2), <math>X \rightarrow Z</math> (5), <math>Y \rightarrow Z</math> (1), <math>Y \rightarrow W</math> (2), <math>Z \rightarrow W</math> (3), <math>W \rightarrow T</math> (1), <math>Z \rightarrow T</math> (8). Qual é o caminho de <b>menor custo</b> de <b>X</b> até <b>T</b>, usando o algoritmo de Dijkstra?</p> <p>a) <math>X \rightarrow Z \rightarrow T</math></p> <p>b) <math>X \rightarrow T</math></p> <p>c) <math>X \rightarrow Y \rightarrow Z \rightarrow T</math></p> <p>d) <math>X \rightarrow Y \rightarrow Z \rightarrow W \rightarrow T</math></p> <p>e) <math>X \rightarrow Y \rightarrow W \rightarrow T</math></p>	
<p>07) Qual das alternativas descreve corretamente uma diferença entre a Busca em Largura (BFS) e a Busca em Profundidade (DFS) em grafos?</p> <p>a) A BFS utiliza uma pilha para gerenciar os vértices, enquanto a DFS usa uma fila.</p> <p>b) A DFS garante encontrar o caminho mais curto entre dois vértices, enquanto a BFS não.</p> <p>c) A BFS explora os vértices por níveis (camadas), enquanto a DFS segue caminhos até o final antes de retroceder.</p> <p>d) Ambas sempre encontram o mesmo caminho, pois percorrem todos os vértices na mesma ordem.</p> <p>e) A DFS é mais adequada que a BFS para encontrar o menor caminho em grafos ponderados.</p>	
<p>08) Qual é o princípio fundamental de um algoritmo guloso?</p> <p>a) Armazenar os resultados de subproblemas para evitar recálculos.</p> <p>b) Tomar a melhor decisão local na esperança de que essa escolha leve a uma solução ótima global.</p> <p>c) Dividir o problema em subproblemas menores e resolver cada um recursivamente.</p> <p>d) Explorar todas as combinações possíveis para encontrar a solução ótima.</p> <p>e) Multiplicar o problema em subproblemas difíceis.</p>	
<p>09) Qual é a principal ideia por trás da técnica de Programação Dinâmica na resolução de problemas?</p> <p>a) Resolver subproblemas repetidamente, mesmo que já tenham sido resolvidos.</p> <p>b) Usar uma tabela para computar posição (x,y) do jogo batalha naval.</p> <p>c) Utilizar estruturas de dados como pilhas e filas para gerenciar a ordem de execução.</p> <p>d) Utilizar laços infinitos para problemas insolúveis.</p> <p>e) Armazenar soluções de subproblemas em uma tabela para evitar recomputações redundantes.</p>	
<p>10) Qual é o principal objetivo de um <b>fluxograma</b> na representação de um algoritmo?</p> <p>a) Escrever o código-fonte final de um programa.</p> <p>b) Testar a eficiência de um algoritmo.</p> <p>c) Ocultar a complexidade do algoritmo.</p> <p>d) Representar graficamente a sequência lógica de operações de um algoritmo.</p> <p>e) Mostrar a beleza visual de uma programação.</p>	
<p>11) Se <math>A &lt; B</math> e <math>B &lt; C</math>, então:</p> <p>a) <math>A &gt; C</math></p> <p>b) <math>A &gt; B</math></p> <p>c) <math>A + K &gt; C + K</math></p> <p>d) <math>C + K &gt; B + K</math></p> <p>e) <math>B &gt; C</math></p>	

Q	A	B	C	D	E		Q	A	B	C	D	E		Q	A	B	C	D	E		Q	A	B	C	D	E	
1							4							7								10					
2							5							8								11					
3							6							9								SIM.					