



Disciplina Técnicas Avançadas de Programação	Curso Sistemas de Informação	Período 5º
Professor Kleber Jacques F. de Souza (klebersouza@pucminas.br)		

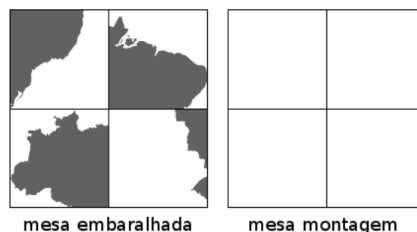
Exercícios - Tentativa e Erro

Instruções

- Esta lista de exercícios deve ser entregue individualmente.
- Esta lista deve ser manuscrita.

Questões

1. Apresente um esboço do esquema geral da Técnica de Projeto de Algoritmos Tentativa e Erro.
2. Considere os algoritmos baseados em Tentativa e Erro. Quando eles são utilizados? Qual é o principal problema que enfrentam os algoritmos que realizam Tentativa e Erro?
3. Considere o problema de montar um quebra-cabeça. O quebra-cabeça possui 4 peças e está embaralhado numa mesa (2x2) espaços, conforme demonstrado na figura abaixo. O objetivo é tentar montá-lo de forma correta posicionando peça por peça em um outro espaço de 2x2 peças, que inicialmente está vazio.



Apresente um algoritmo que utiliza a abordagem de Tentativa e Erro para resolver este problema.

Resposta:

Entrada: Mesa1[2x2] com 4 peças, Mesa2[2x2] vazia

Algoritmo:

- 1 Para cada peça i [1..4] da Mesa1.
 - 1.1 Colocar a peça i na próxima posição vazia na Mesa2
 - 1.2 Verifica se a peça i está na sua posição correta
 - 1.3 Se sim Repete 1.1 para a próxima peça i
 - 1.4 Senão Tentar colocar a peça i em outra posição vazia repete passo 1.2
- 2 Finaliza quando todas as peças estiverem na sua posição correta.

Saída: Quebra cabeça montado.

4. Considere o seguinte problema: Temos uma matriz de 3×3 e devemos colocar em cada posição uma letra do alfabeto de tal maneira que todas linhas e colunas formem uma palavra que existe em português. Eis um exemplo de solução para esse problema:

p	u	s
o	v	o
s	a	l

Apresente um algoritmo que utiliza a abordagem de Tentativa e Erro para resolver este problema.

Resposta:

Entrada: matriz de 3×3

Algoritmo:

- 1 Para cada posição i $[0..3]$ e j $[0..3]$ da matriz.
- 2 Preencher a posição $(0,0)$ da matriz com uma letra de A até Z, realizando todas as combinações possíveis.
- 3 Preencher as próximas posições da **linha** da matriz apenas com as letras de A até Z que possam complementar as letras já preenchidas para formarem uma palavra que existe no português. (Não Realizar todas as combinações!)
- 4 Preencher as próximas posições da **coluna** da matriz apenas com as letras de A até Z que possam complementar as letras já preenchidas para formarem uma palavra que existe no português. (Não Realizar todas as combinações!)
- 5 Repetir o passo 3 e 4 para as próximas posições vazias.
- 6 Para cada combinação gerada.
- 6.1 Verifica se todas linhas e colunas formam uma palavra que existe em português. Se sim retorna a matriz, senão passa para a próxima combinação.

Saída: retorna Matriz com as letras onde todas linhas e colunas formem uma palavra que existe em português.

5. Um caixeiro viajante deseja visitar N cidades e entre cada par de cidades existe uma rota. Cada rota possui uma distância (ou o custo necessário) para percorrê-la. O caixeiro viajante deseja encontrar:
 - um caminho que passe por cada cidade apenas uma vez,
 - o caminho que tenha o menor custo possível.

Apresente um algoritmo que utiliza a abordagem de Tentativa e Erro para resolver este problema.

Resposta:

Entrada: Cidades $[1..N]$

Algoritmo:

- 1 Define $CUSTO_MELHOR_ROTA = \infty$
- 2 Para cada cidade i $[1..n]$
 - 2.1 Realizar todas as rotas que passe por todas as outras cidades apenas uma vez, a partir da cidade i .
 - 2.2 Para cada cidade visitada na rota, verificar se o custo atual é maior que $CUSTO_MELHOR_ROTA$, se sim parar a execução da rota atual e passar para outra rota.
 - 2.3 Ao finalizar uma rota, se o custo da rota for menor que $CUSTO_MELHOR_ROTA$, ele se torna a melhor rota e $CUSTO_MELHOR_ROTA = \text{custo da rota atual}$.

Saída: melhor rota, com o menor custo possível.