

86

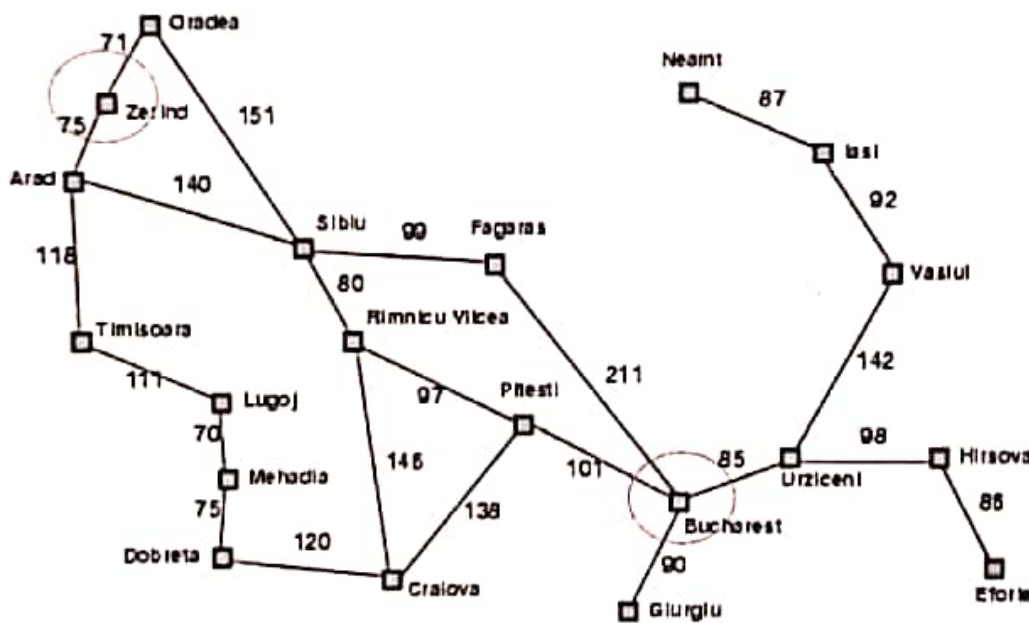
**1ª Avaliação Parcial**  
 Curso: Engenharia de Computação  
 Disciplina: Inteligência Computacional  
 Prof. Jarbas Joaci de Mesquita Sá Junior  
 UFC – Universidade Federal do Ceará

Nome: \_\_\_\_\_

Data 27/03/2019

1. Explique como funcionam as buscas em extensão, profundidade e aprofundamento iterativo em profundidade. Além disso, defina as vantagens e desvantagens dessas buscas em termos de **completeza**, **otimalidade**, **complexidade de tempo** (quantidade de nodos gerados) e **complexidade de espaço** (quantidade de nodos armazenados na memória). (2.0 pontos)

2. Considerando o mapa da *Romênia* abaixo e a relação das distâncias em linha reta de cada cidade em relação à *Bucharest* (considerar essas distâncias a heurística  $h(n)$  do problema), responda às seguintes solicitações:



Straight-line distance to Bucharest	
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

a) Considerando o estado inicial a cidade de *Zerind* e o estado meta a cidade de *Bucharest*, desenhe as fases de uma busca gulosa pela melhor escolha. (2.0 pontos)

b) Com os mesmos estados inicial e meta do item anterior, desenhe as fases de uma busca A\*. (2.0 pontos)

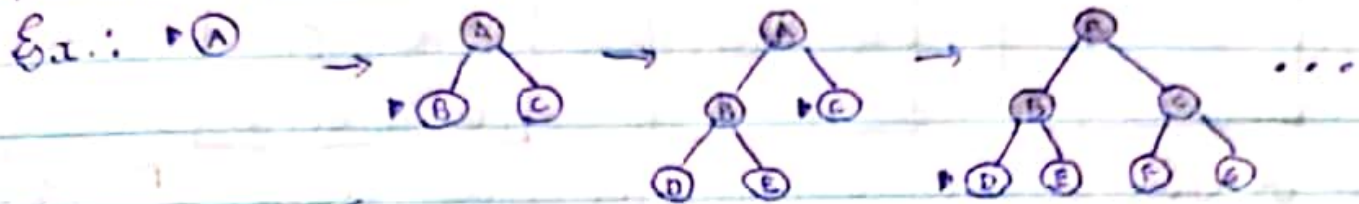
3. Explique as buscas *hill climbing* (subida de encosta), de *têmpera simulada* e em *feixe local*. (2.0 pontos)

4. O que significa um agente de inferência lógica ser **consistente** e **completo**? Quais as **limitações** da inferência em lógica proposicional? (2.0 ponto)

*simples demais para o mundo real*

### ① - Busca em largura (extensão):

Sempre analisa as camadas mais rasas antes de passar para a próxima:



■ Completude: É completa, contanto que a árvore seja finita

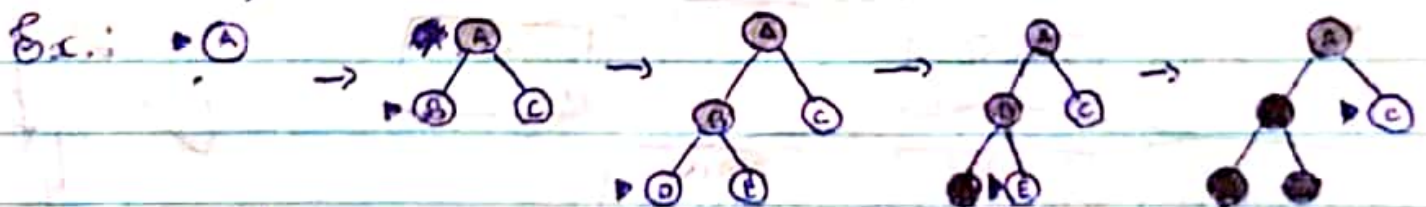
■ Otimalidade: É ótima, com a condição que os custos dos passos sejam maiores ou iguais aos custos dos camadas anteriores.

■ Complexidade (tempo): Exponencial  $[O(b^{d+1})]$

■ Complexidade (espaço): Exponencial  $[O(b^{d+1})]$

### - Busca em profundidade:

Sempre analisa os nós mais profundos, ainda não analisados, da árvore.



■ Completude: não é completa, pois pode ficar presa em ramificações sem nunca encontrar o objetivo

■ Otimalidade: não é ótima, pois pode existir melhores soluções em nós ainda não explorados.

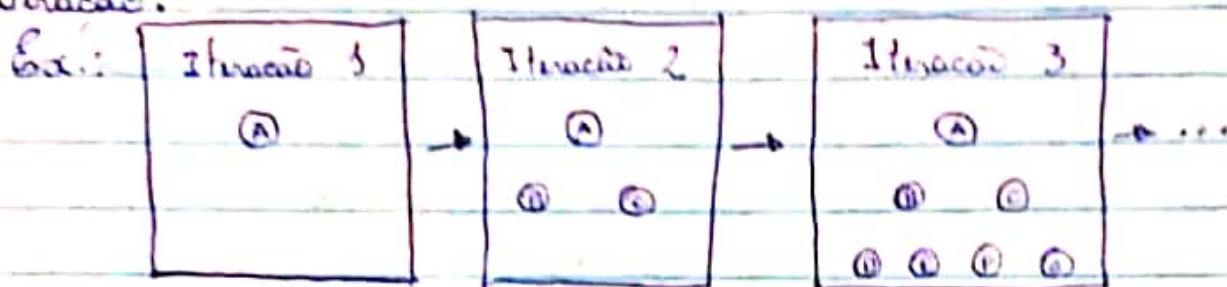
■ Complexidade (tempo): Exponencial  $[O(b^m)]$

■ Complexidade (espaço): Linear  $[O(bm)]$



- Busca em profundidade iterativa:

Tenta encontrar o melhor resultado a partir de colocar um limite na árvore e aumentar esse limite a cada iteração.

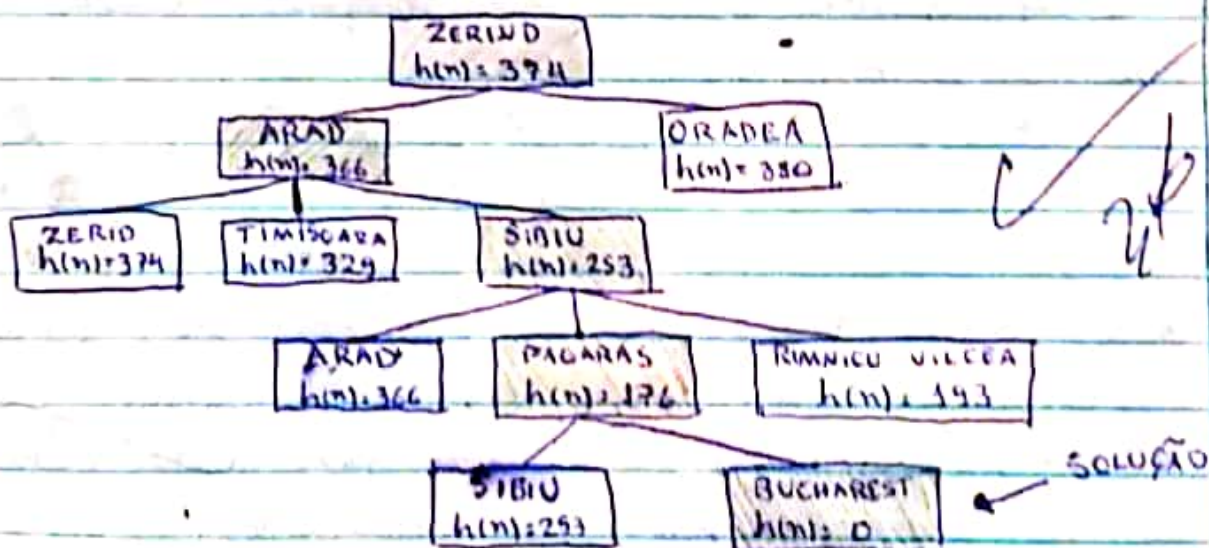


Obs.: A cada iteração é realizada uma busca em profundidade conforme explicado anteriormente.

- Completude: Completa.
- Otimalidade: Ótima.
- Complexidade (tempo): Exponencial  $[O(b^d)]$
- Complexidade (espaço): Linear  $[O(bd)]$

②

a) Busca Gulosa



PASSO-A-PASSO

1 - TESTAMOS SE ZERID É OBJETIVO, COMO NÃO É, GERAMOS OS FILHOS

2 - O MENOR VALOR DE HEURISTICA ENTRE OS <sup>DOIS</sup> FILHOS É DE ARAD, SEGUIMOS PARA ELE, VERIFICAMOS QUE NÃO É OBJETIVO E GERAMOS SEUS FILHOS.

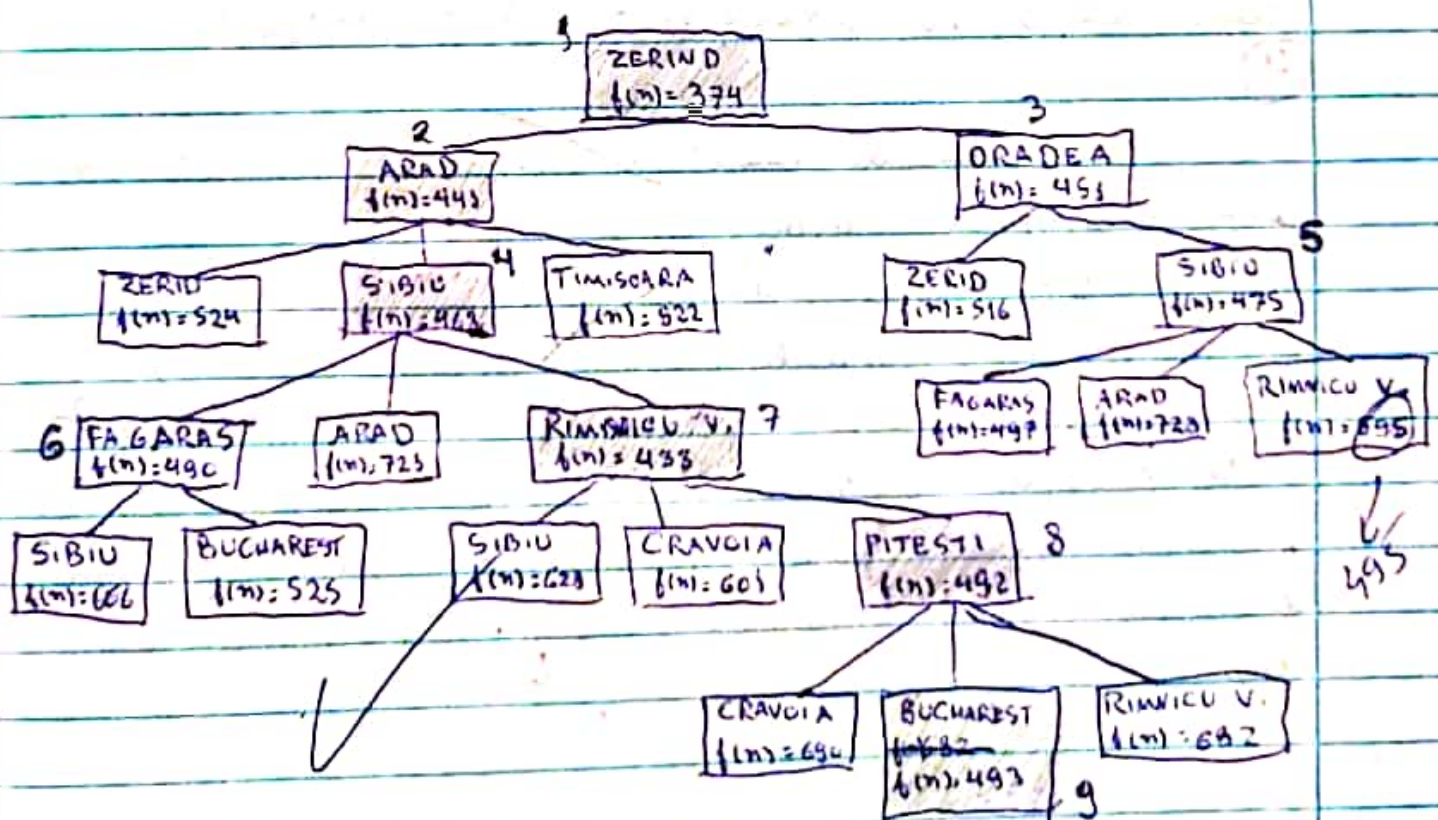
3 - SEGUIMOS ENTÃO PARA SIBIU QUE É O QUE TEM A MENOR HEURISTICA, NÃO É OBJETIVO, LOGO GERAMOS SEUS FILHOS

4 - SEGUIMOS PARA FAGARAS, TAMBÉM VERIFICAMOS QUE NÃO É OBJETIVO, ENTÃO GERAMOS OS FILHOS.

5 - ENFIM, TEMOS HEURISTICA = 0 PARA BUCHAREST, E AO VERIFICAR, PERCEBEMOS QUE É O OBJETIVO E A BUSCA FINALIZA

ZERIND → ARAD → SIBIU → FAGARAS → BUCHAREST //

b) busca A\* ( $f(n) = h(n) + g(n)$ )



PASSO 1 - ZERIND NÃO É OBJETIVO, CRIOU FILHOS

2 - FOI PRA ARAD, NÃO É OBJETIVO, CRIOU FILHOS

3 - ORADEA TINHA O <sup>MENOR</sup> MAIOR VALOR DA FUNÇÃO



- ③ A Busca Hill climbing é ~~uma~~ uma estratégia gulosa que busca maximizar ou minimizar um valor através de uma busca local. em hill climbing dado um valor inicial, buscamos pela vizinhança pontos que possam ser melhores do que o que estamos. A busca para quando não há uma solução melhor na vizinhança.
- Em Tempera simulada utilizamos a estratégia de fazer randomicamente decisões ruins em busca de um ponto máximo (ou mínimo) global. Fazendo isso diminuimos aos poucos a frequência (consequentemente a amplitude) até encontrarmos o melhor ponto.

- Em busca local, adotamos a mesma estratégia da busca hill climbing original, com a exceção que o ~~ponto~~ o ponto inicial é explorado com  $K$  pontos simultaneamente. Se encontrarmos a meta, escolhemos o  $K$  que é o melhor, quando não, redamos novamente o algoritmo com os  $K$  melhores sucessores. Os pontos iniciais são escolhidos randomicamente.

4/3

FOI PARA ORADEA, NÃO É OBJETIVO CRIOU FILHOS  
4 - SIBIU MENOR VALOR, NÃO É OBJETIVO, CRIOU FILHOS.

5 - SIBIU EM OUTRA RAMIFICAÇÃO ERA MENOR VALOR, NÃO É OBJETIVO, CRIOU FILHOS

6 - FAGARAS EM OUTRA RAMIFICAÇÃO FOI MENOR VALOR, NÃO É OBJETIVO, CRIOU FILHOS

7 - RIMNICU VILCEA MENOR VALOR, NÃO É OBJETIVO, CRIOU FILHOS

8 - PITESTI MENOR VALOR, NÃO É OBJETIVO, CRIOU FILHOS.

9 - BUCHAREST COM MENOR VALOR, OBJETIVO, FIM DA BUSCA

ZERIND → ARAD → SIBIU → RIMNICU V. → PITESTI → BUCHAREST

(4)

CONSISTENTE - Quando ~~uma~~ <sup>todas</sup> as inferências derivadas de um BC não são verdadeiras.

COMPLETO - Quando ~~as~~ <sup>todas</sup> inferências que pertencem a um BC, são verdadeiras ~~de~~ <sup>de</sup> ~~daquele~~ <sup>de</sup> BC.

As principais limitações são que a lógica proposicional é simples demais para representar o mundo real, <sup>2</sup> ~~isto~~ problemas muito complexos, exigem um número muito grande de inferências.