Resolução de Problemas por meio de busca – Busca Heurística

Aula 03 – Busca Heurística

Prof. Msc. Luiz Mário Lustosa Pascoal

Estratégias de Busca Exaustiva (Cega)

- Encontram soluções para problemas pela geração *sistemática* de novos estados, que são comparados ao objetivo;
- São *ineficientes* na maioria dos casos:
  - utilizam apenas o *custo de caminho* do nó atual ao nó inicial (função *g*) para decidir qual o próximo nó da fronteira a ser expandido.
  - essa medida nem sempre conduz a busca na direção do objetivo.
- Como encontrar um barco perdido?
  - não podemos procurar no oceano inteiro...
  - observamos as correntes marítimas, o vento, etc...

Estratégias Busca Heurística (Informada)

- Utilizam conhecimento específico do problema na escolha do próximo nó a ser expandido
  - barco perdido
    - correntes marítimas, vento, etc...
- Aplicam de uma função de avaliação a cada nó na fronteira do espaço de estados
  - essa função estima o custo de caminho do nó atual até o objetivo mais próximo utilizando uma função heurística.
  - Função heurística
    - estima o custo do caminho mais barato do estado atual até o estado final mais próximo.

Busca com informação (ou heurística)

- Utiliza conhecimento específico sobre o problema para encontrar soluções de forma mais eficiente do que a busca cega.
  - Conhecimento específico além da definição do problema.
- Abordagem geral: busca pela melhor escolha.
  - Utiliza uma função de avaliação para cada nó.
  - Expande o nó que tem a função de avaliação mais baixa.
  - Dependendo da função de avaliação, a estratégia de busca muda.

Busca Heurística

- Classes de algoritmos para busca heurística:
  1. Busca pela melhor escolha (*Best-First search*)
  2. Busca com limite de memória

Busca pela Melhor Escolha
Best-First Search

- Busca pela Melhor Escolha - BME
  - Busca genérica onde o nó de menor custo "aparente" na fronteira do espaço de estados é expandido primeiro
- Duas abordagens básicas:
  - 1. Busca Gulosa (Greedy search)
  - 2. Algoritmo A*

Busca pela melhor escolha

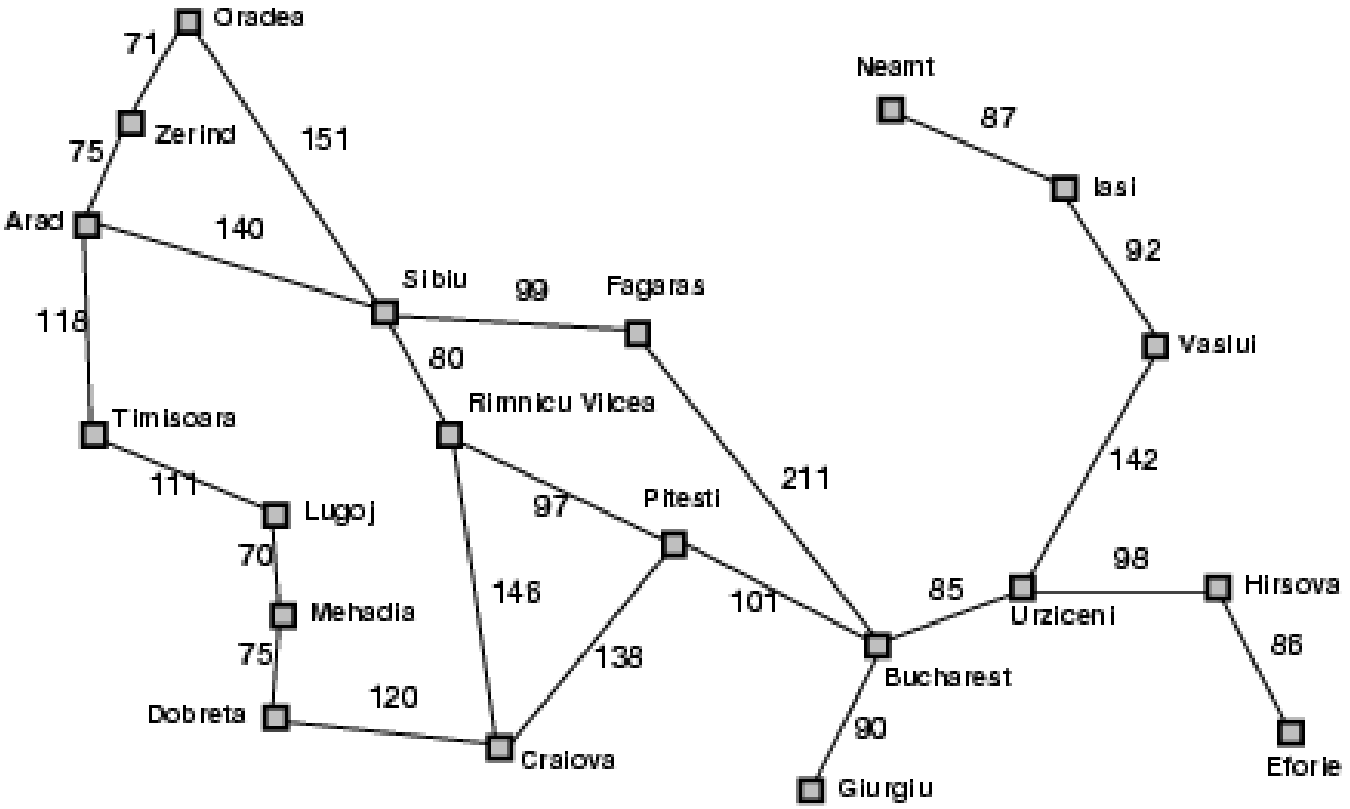
- Idéia: usar uma função de avaliação *f(n)* para cada nó.
  - estimativa do quanto aquele nó é desejável
  - → Expandir nó mais desejável que ainda não foi expandido

- Implementação:

  Ordenar nós na borda em ordem decrescente de acordo com a função de avaliação

- Casos especiais:
  - Busca gulosa pela melhor escolha
  - Busca A*

Romênia com custos em km



Distância em linha reta para Buchrest

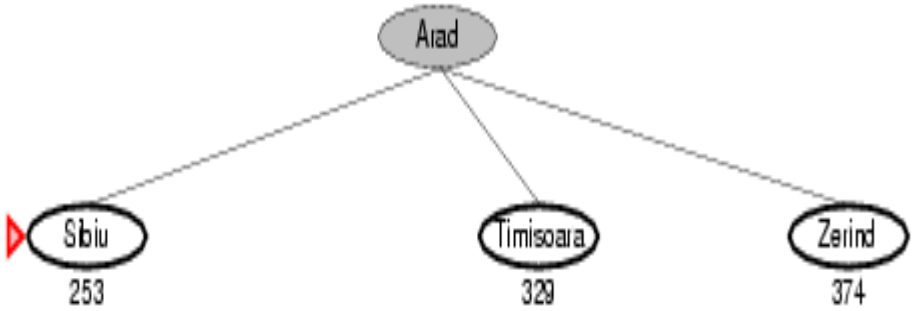
Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	10
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Busca gulosa pela melhor escolha

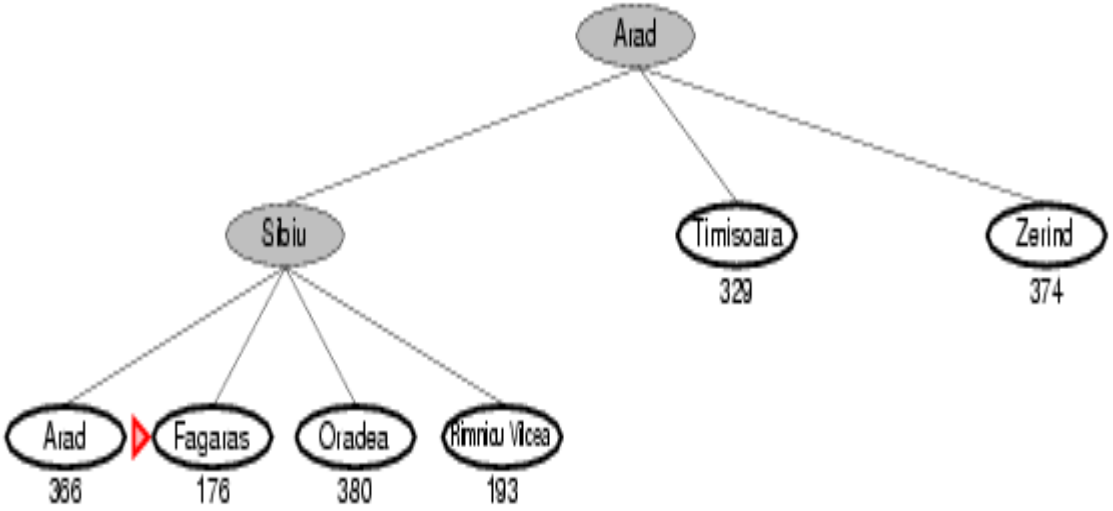
- Função de avaliação $f(n) = h(n)$ (heurística)
  = estimativa do custo de n até o objetivo
  ex., $h_{DLR}(n)$ = distância em linha reta de n até Bucareste.
- Busca gulosa pela melhor escolha expande o nó que parece mais próximo ao objetivo de acordo com a função heurística.

Exemplo de busca gulosa pela melhor escolha

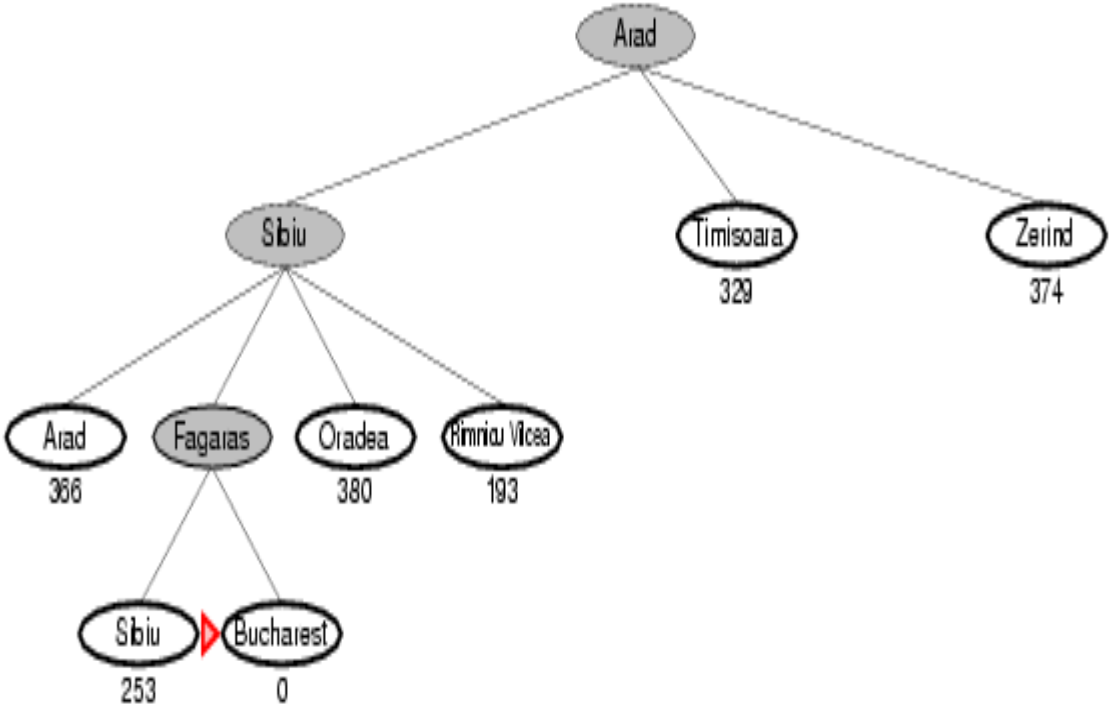
Exemplo de busca gulosa pela melhor escolha



Exemplo de busca gulosa pela melhor escolha



Exemplo de busca gulosa pela melhor escolha



Busca gulosa pela melhor escolha

- Não é ótima, pois segue o melhor passo **considerando somente o estado atual.**
  - Pode haver um caminho melhor seguindo algumas opções piores em alguns pontos da árvore de busca.
    - No exemplo, a Busca Gulosa escolhe o caminho que é mais econômico via Fagaras, porém existe um caminho mais curto via Rimnicu Vilcea.
- Minimizar h(n) é suscetível a falsos inícios.
  - Ex. Ir de Iasi a Fagaras
    - Heurística sugerirá ir a Neamt, que é um beco sem saída.
    - Se repetições não forem detectadas a busca entrará em loop.

Propriedades da busca gulosa pela melhor escolha

- Completa? Não – pode ficar presa em loops, ex., Iasi → Neamt → Iasi → Neamt
- Tempo? $O(b^m)$ no pior caso, mas uma boa função heurística pode levar a uma redução substancial
- Espaço? $O(b^m)$ – mantém todos os nós na memória
- Ótima? Não

Busca A*

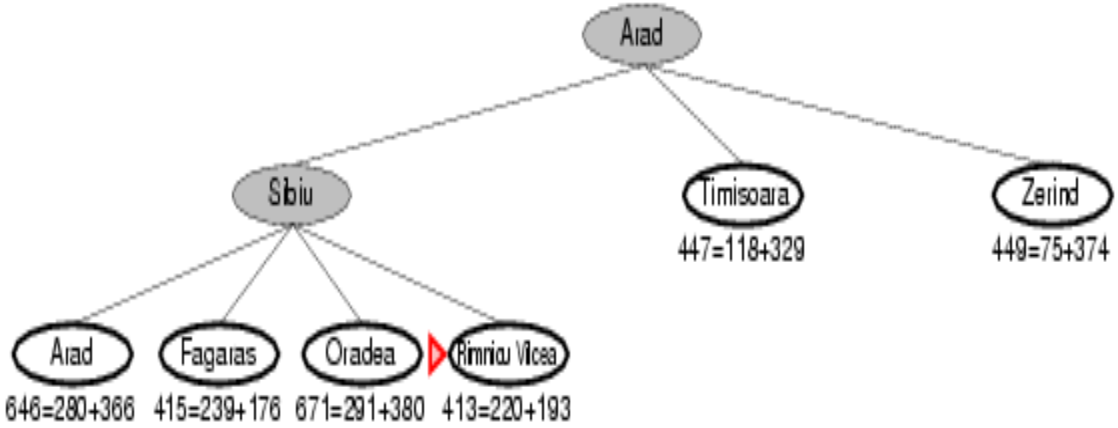
- **Idéia:** Utiliza tanto o custo do caminho realizado até o momento, ***g(n)***, quanto o valor da heurística, ***h(n)***, para decisão de expansão do nó.

- Função de avaliação: ***f(n) = g(n) + h(n)***
  - *g(n)* = custo até o momento para alcançar *n*
  - *h(n)* = custo estimado de *n* até o objetivo
  - *f(n)* = custo total estimado do caminho através de *n* até o objetivo.

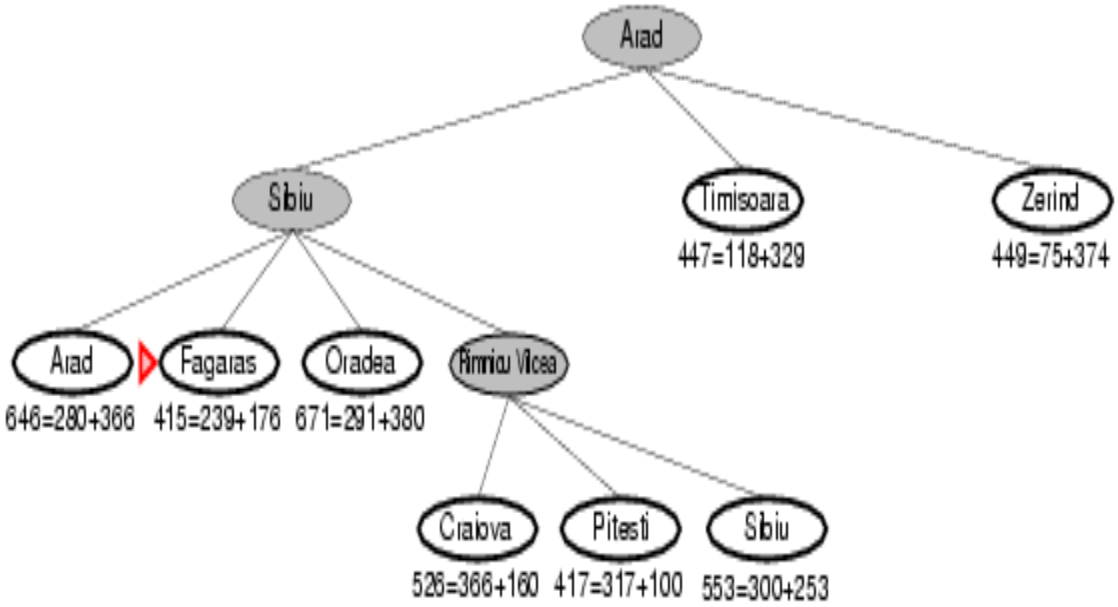
Exemplo de busca A*

Exemplo de busca A*

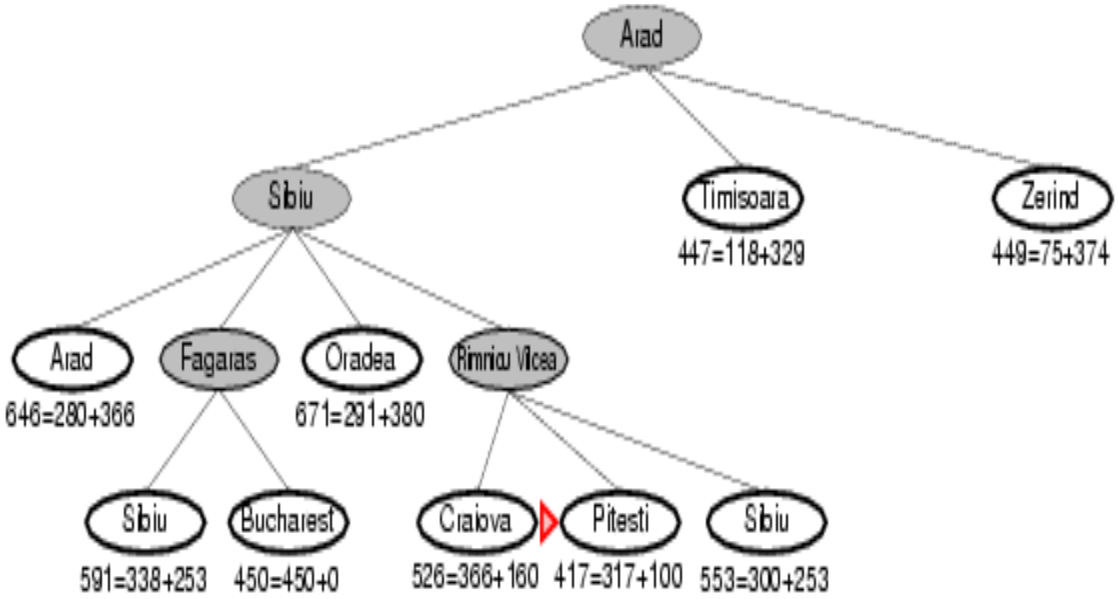
Exemplo de busca A*



Exemplo de busca A*



Exemplo de busca A*



Exemplo de busca A*

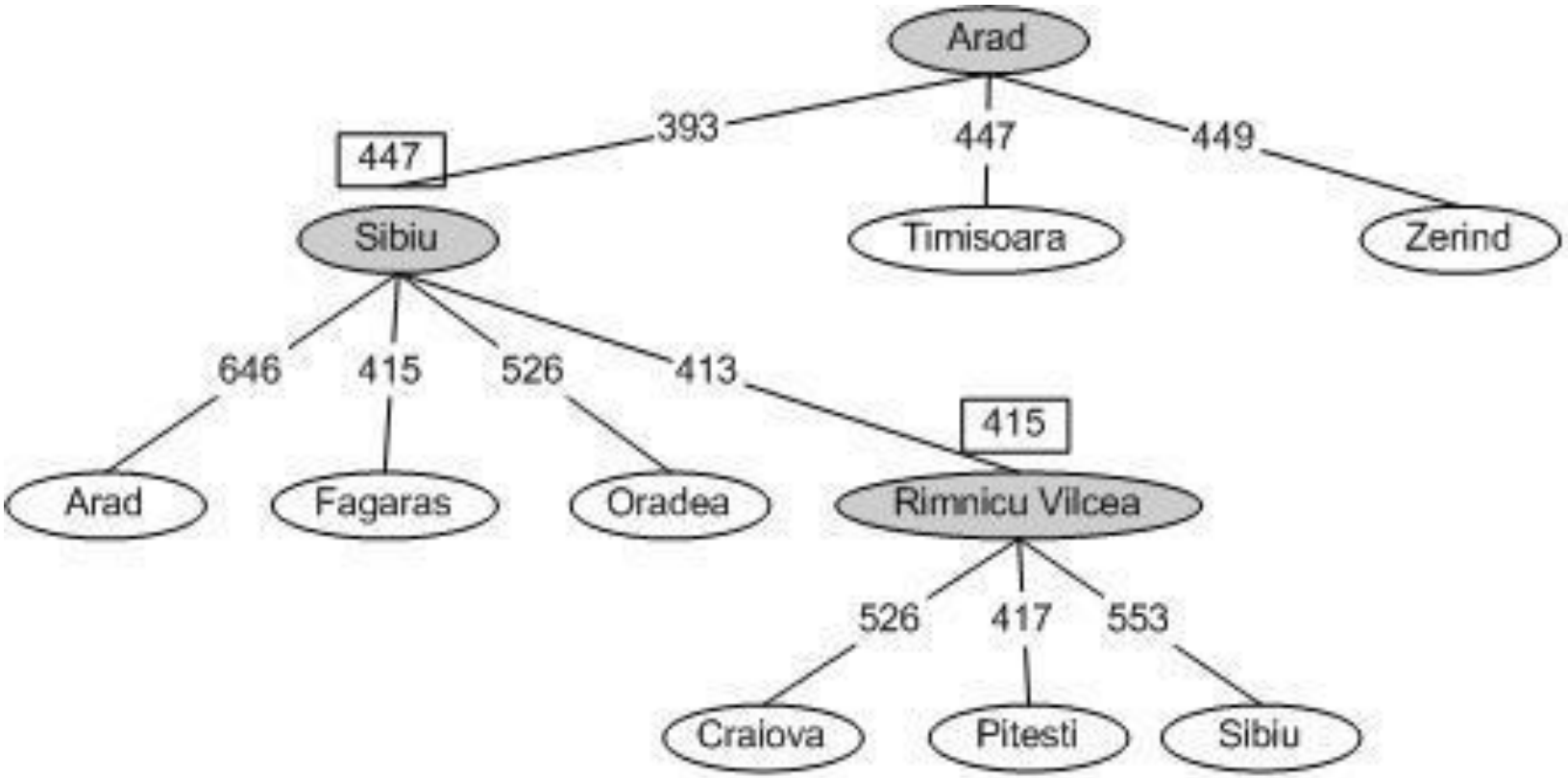
Propriedades da Busca A*

- Completa? Sim (a não ser que exista uma quantidade infinita de nós com $f \leq f(G)$)
- Tempo? Exponencial no pior caso
- Espaço? Mantém todos os nós na memória: g(n)
  - Possibilita o backtracking.
- Ótima? Sim
- Otimamente eficiente
  - Nenhum outro algoritmo de busca ótimo tem garantia de expandir um número de nós menor que A*. Isso porque qualquer algoritmo que não expande todos os nós com f(n) < C* corre o risco de omitir uma solução ótima.

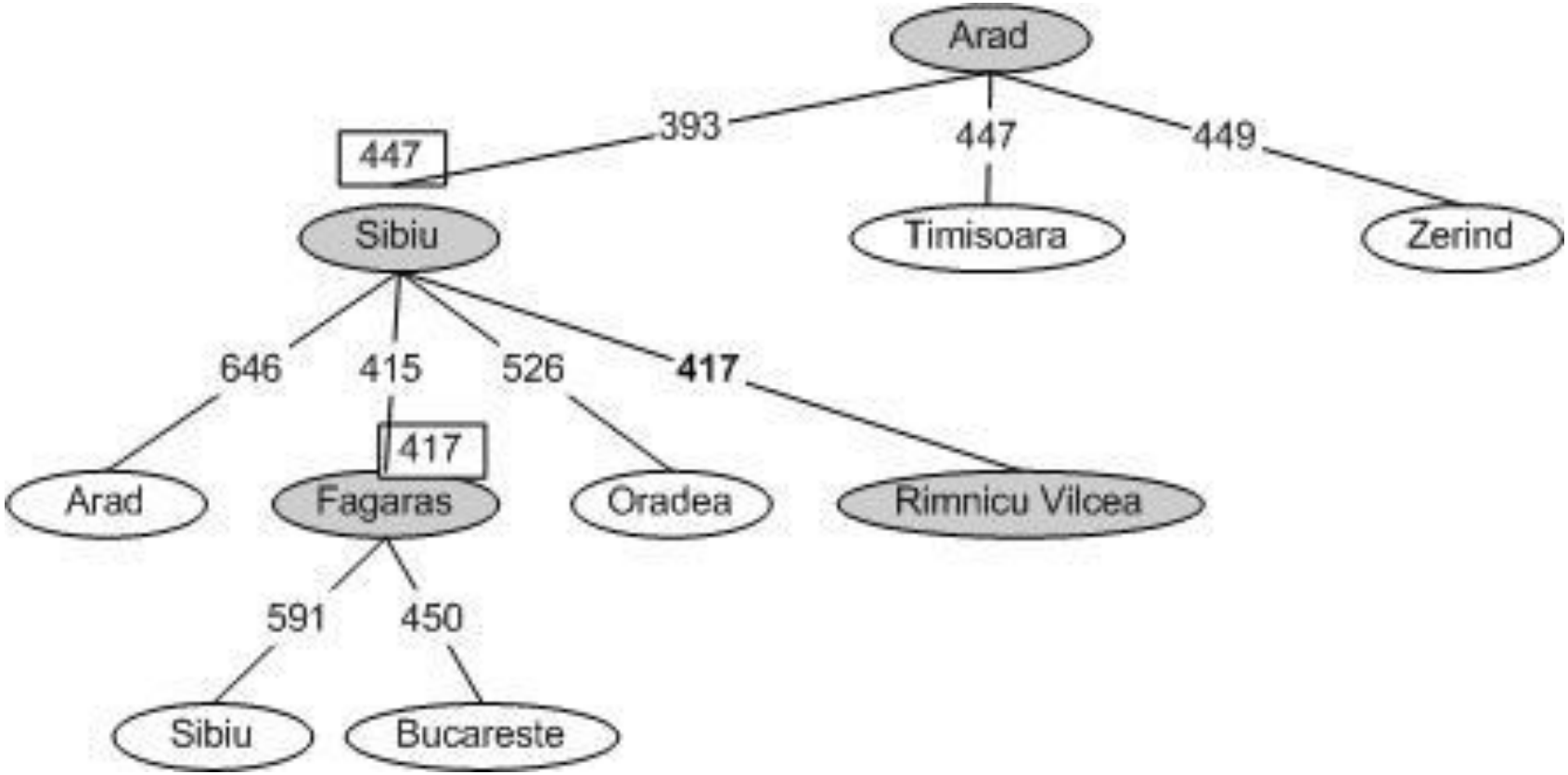
Busca Recursiva pelo melhor (BRPM)

- Tenta imitar a **busca pela melhor escolha-padrão**
- Utiliza espaço **linear**
- É semelhante à busca em profundidade recursiva, mas **guarda o valor de f do melhor caminho alternativo**
- Se o custo do nó atual exceder f, a recursão retorna ao caminho alternativo
- Repõe o valor de f de cada nó ao longo do caminho com o melhor valor de f de seus filhos
- Podendo decidir se vale a pena voltar a expandir uma árvore esquecida

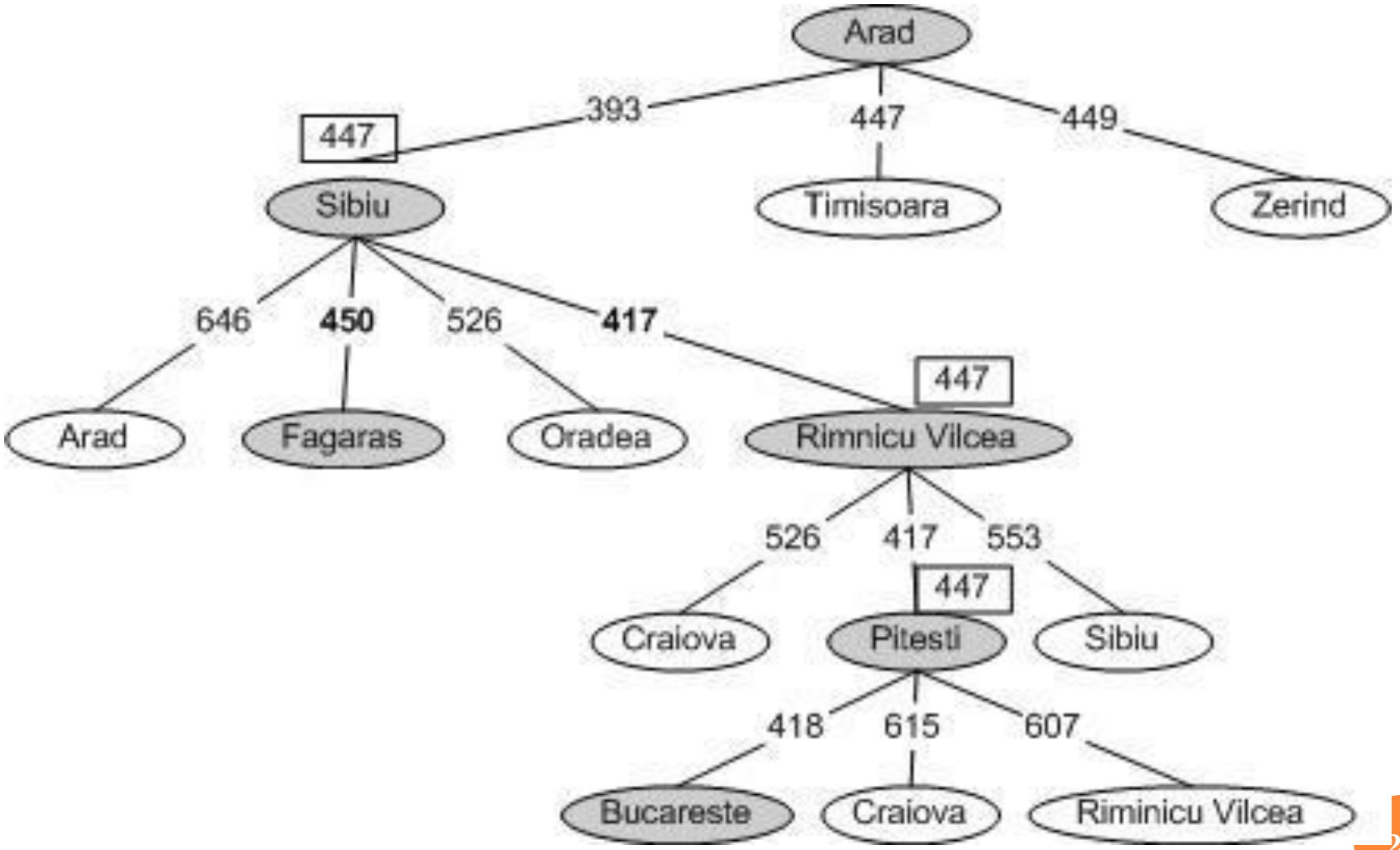
Busca Recursiva pelo melhor (BRPM)



Busca Recursiva pelo melhor (BRPM)



Busca Recursiva pelo melhor (BRPM)



Busca Recursiva pelo melhor (BRPM)

- Ainda sofre da geração excessiva de nós
  - toda vez que o melhor caminho é expandido, há uma boa chance que o valor de f aumente
  - E então um 2º melhor caminho deve ser seguido, reexpandindo os nós esquecidos
- É ótimo se *h(n)* é admissível
- Complexidade de espaço O(*bd*)
- Complexidade de tempo depende:
  - Da exatidão da função heurística
  - Da freqüência com que o melhor caminho muda a medida que os nós são expandidos