

Inteligência Artificial

Estratégias de Busca Parte 3

Prof. Jefferson Moraes

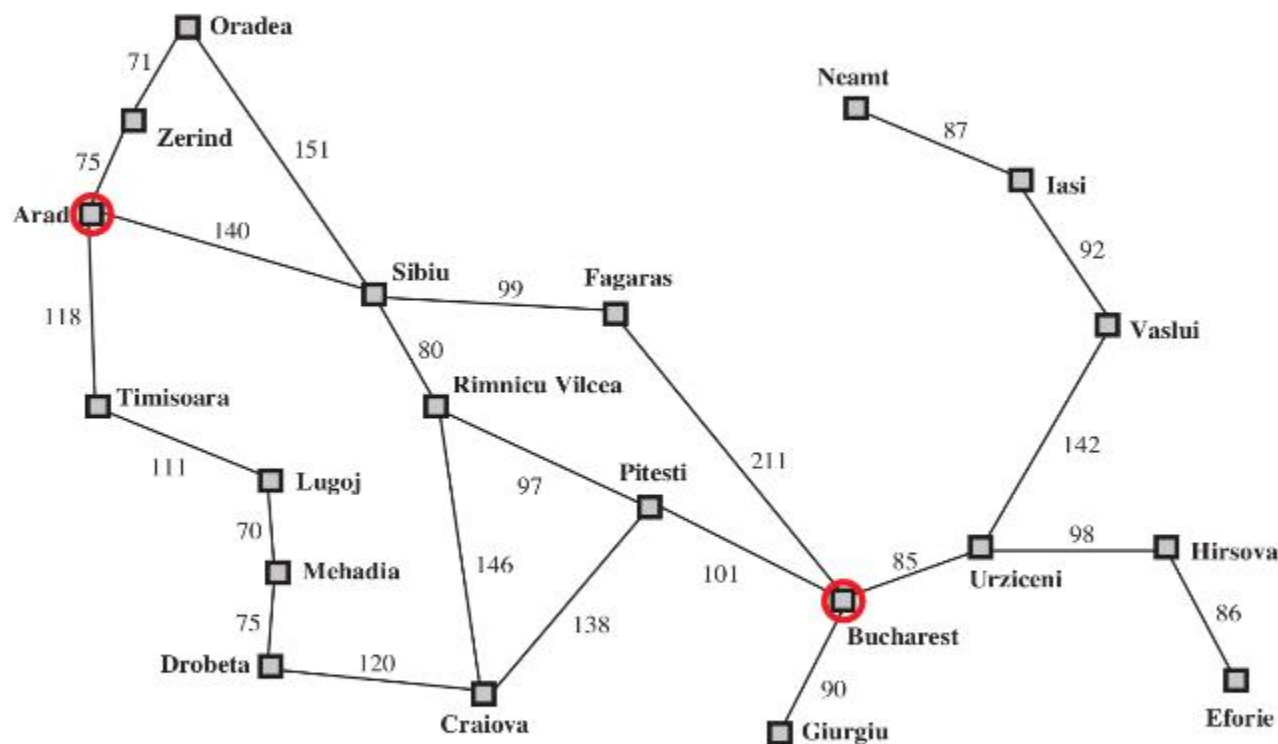
Busca Com Informação

- **Na busca sem informação:** os algoritmos encontram soluções gerando sistematicamente novos estados e comparando-os com o objetivo. Costumam ser ineficientes na maioria dos casos
- **Na busca com informação**
 - Utiliza **conhecimento** (heurística) de um problema específico além da definição do problema em si
 - A heurística auxilia a encontrar os nós mais promissores a cada passo por meio de uma **função** que **estima** distância ao objetivo
 - **Funções heurísticas $f(n)$**
 - São não negativas e de problemas específicos
 - Estima o caminho de menor custo de **n até um nó objetivo**
 - Restrição: se n for um nó objetivo, então $f(n) = 0$

Busca Com Informação - Busca Gulosa

- O fundamento da busca gulosa: espera-se que ao expandir o nó que está mais próximo do objetivo possa conduzir a uma solução rapidamente
- Avalia os nós apenas por uma **função heurística** $f(n)$
 - Cada nó selecionado para expansão é baseado nesta função
 - A escolha de $f(n)$ determina a estratégia de busca
- No caso da busca gulosa, assumiremos uma instância de $f(n) = h(n)$
 - Estima o menor caminho do estado n até o estado objetivo (distância em linha reta)
 - Se n é o estado objetivo, então $h(n) = 0$
 - Se o objetivo for **Bucareste**, então as distâncias serão:

Busca Com Informação - Busca Gulosa



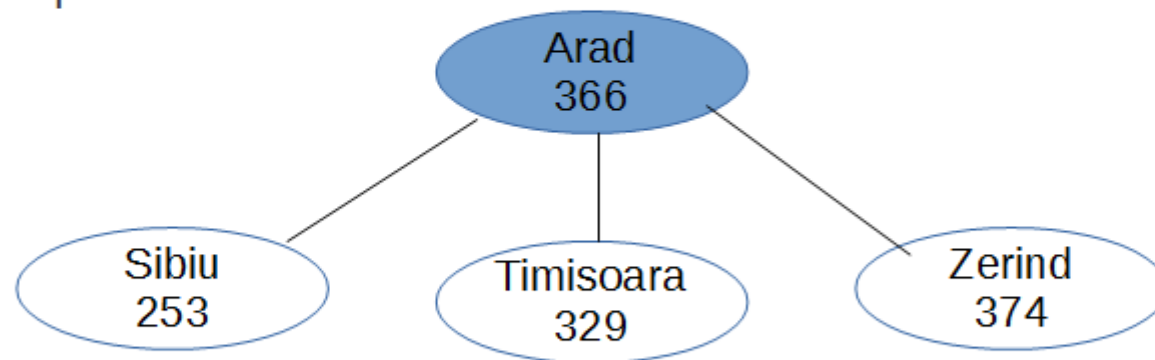
Arad	366
Bucareste	0
Craiova	160
Drobeta	242
Eforie	161
Fagaras	176
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	100
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Busca Com Informação - Busca Gulosa

- O Estado inicial é o Arad



- Expande Arad

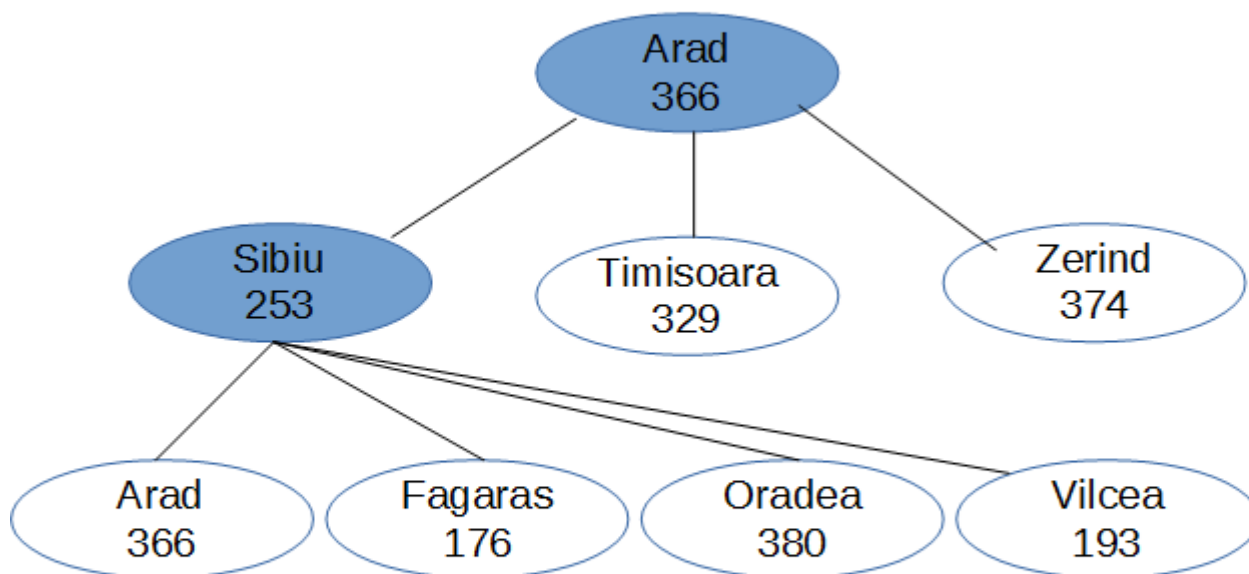


- Qual nó expandir?
 - Sibiu será expandido

Arad	366
Bucareste	0
Craiova	160
Drobeta	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

Busca Com Informação - Busca Gulosa

- Qual o próximo nó a ser expandido?

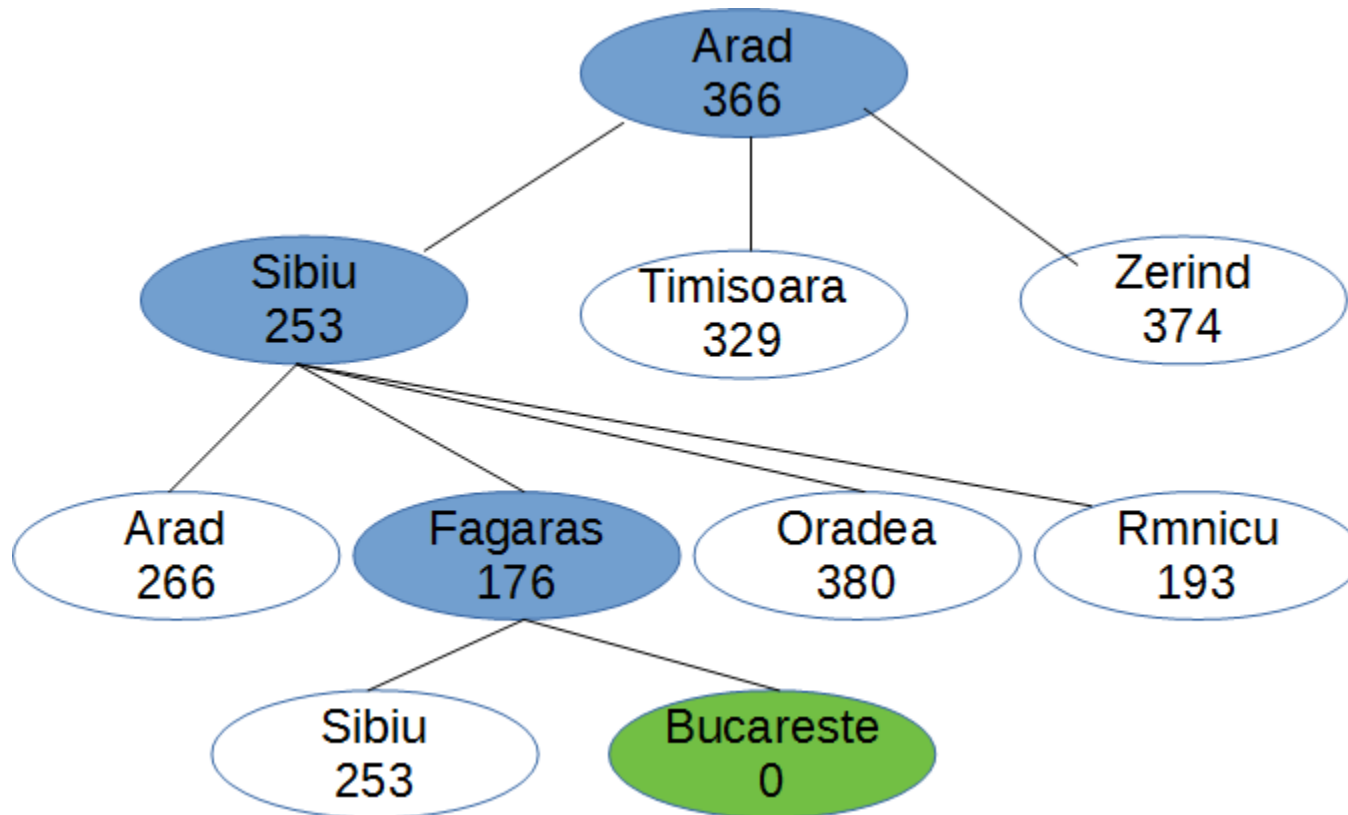


- O próximo nó a ser expandido será o Fagaras

Arad	366
Bucareste	0
Craiova	160
Drobeta	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

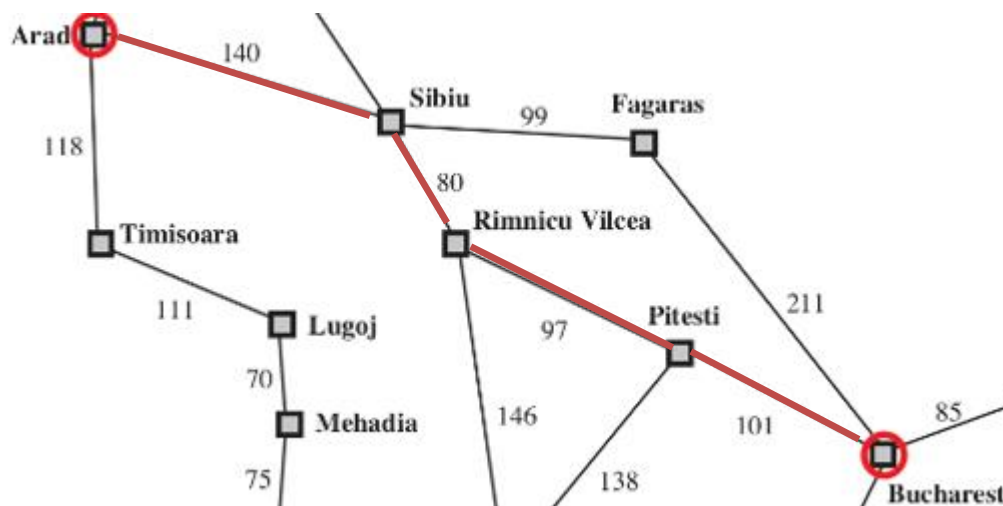
Busca Com Informação - Busca Gulosa

- Fagaras vai gerar Bucareste, que é o objetivo



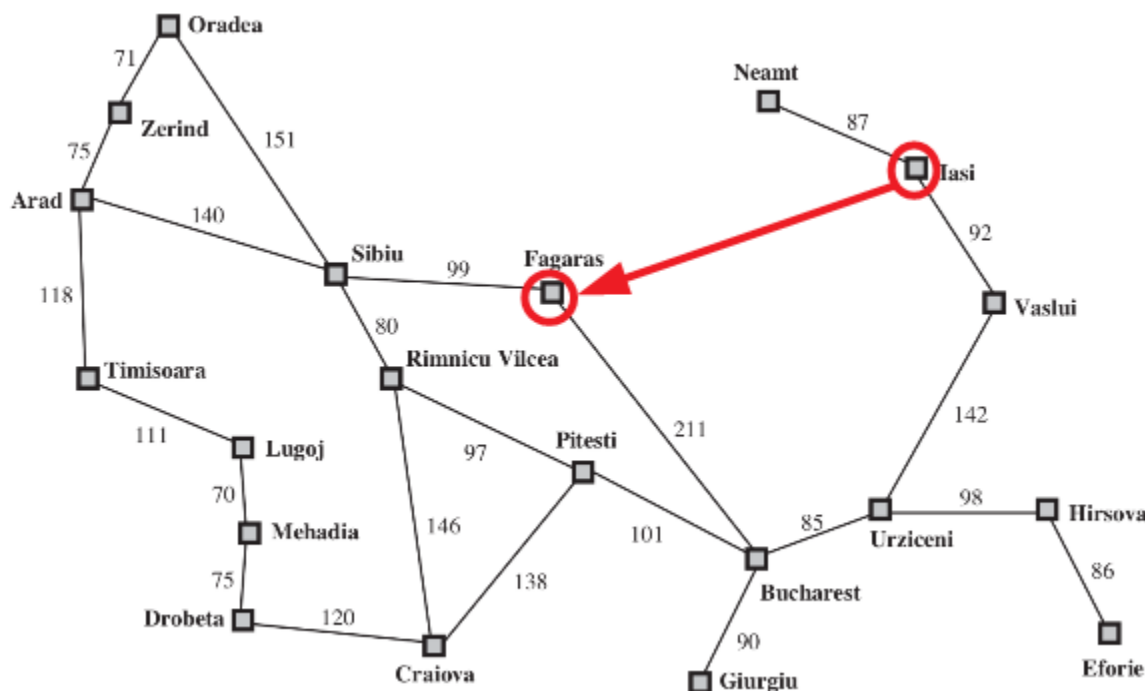
Busca Com Informação - Busca Gulosa

- A busca gulosa, utilizando h , encontrou uma solução, sem nunca expandir um nó que não estiver no caminho da solução
- Contudo, a solução encontrada **não é ótima**
 - O caminho via Sibiu e Fagaras para Bucareste é 32 km mais longo do que por Vilcea e Pitesti
- Isso mostra o porquê do algoritmo ser chamado de **guloso**
 - Escolhe sempre uma resposta ótima local
 - Pode haver um caminho melhor seguindo algumas opções piores em alguns pontos da árvore de busca



Busca Com Informação - Busca Gulosa

- **Completa**: sim (em grafos) e não (em árvores)
 - Pode entrar em caminho infinito (e.g., Iasi → Fagaras)
- **Ótima**: não
- **Complexidade tempo e espaço**: $O(b^m)$, onde m é a profundidade máxima do espaço de busca



Busca Com Informação - Busca A*

- Essa busca avalia os nós através da combinação de
 - $g(n)$: o custo real do **nó inicial** até o **nó n**
 - $h(n)$: o custo estimado para ir do **nó n** até o **nó objetivo**

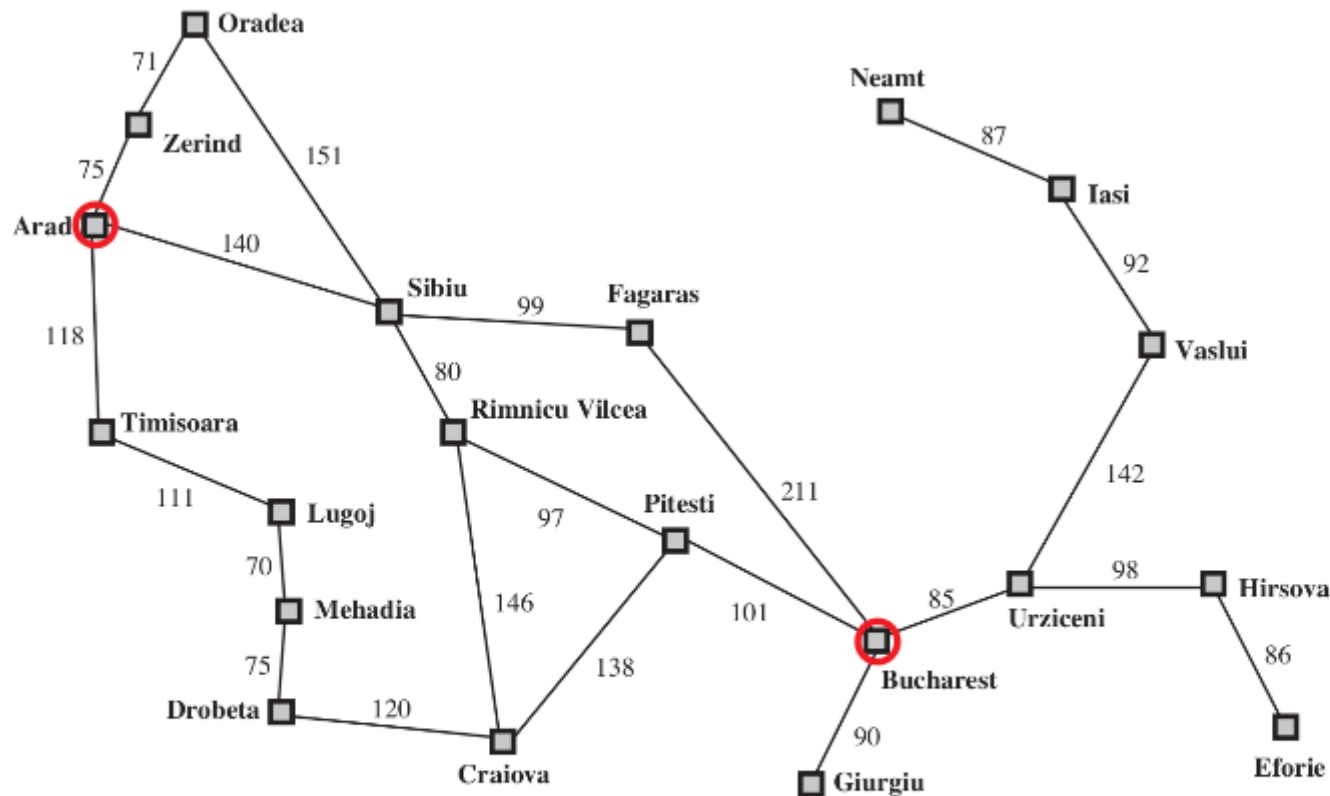
$$f(n) = g(n) + h(n)$$

Função heurística.

O mesmo $g(n)$ da busca de custo uniforme.

- $f(n)$: custo estimado da solução de menor custo através do nó **n**
- O algoritmo é idêntico à busca de custo uniforme, só que usa **$g + h$** em vez de apenas **g**
- Vamos observar o exemplo da Romênia a seguir

Busca Com Informação - Busca A*



Arad	366
Bucareste	0
Craiova	160
Drobeta	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

Busca Com Informação - Busca A*

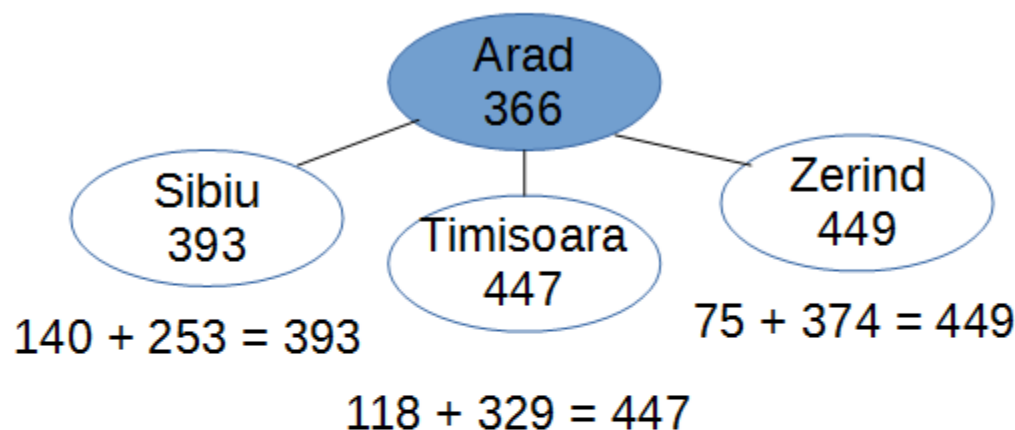
- a) Estado inicial



$$0 + 366 = 366$$

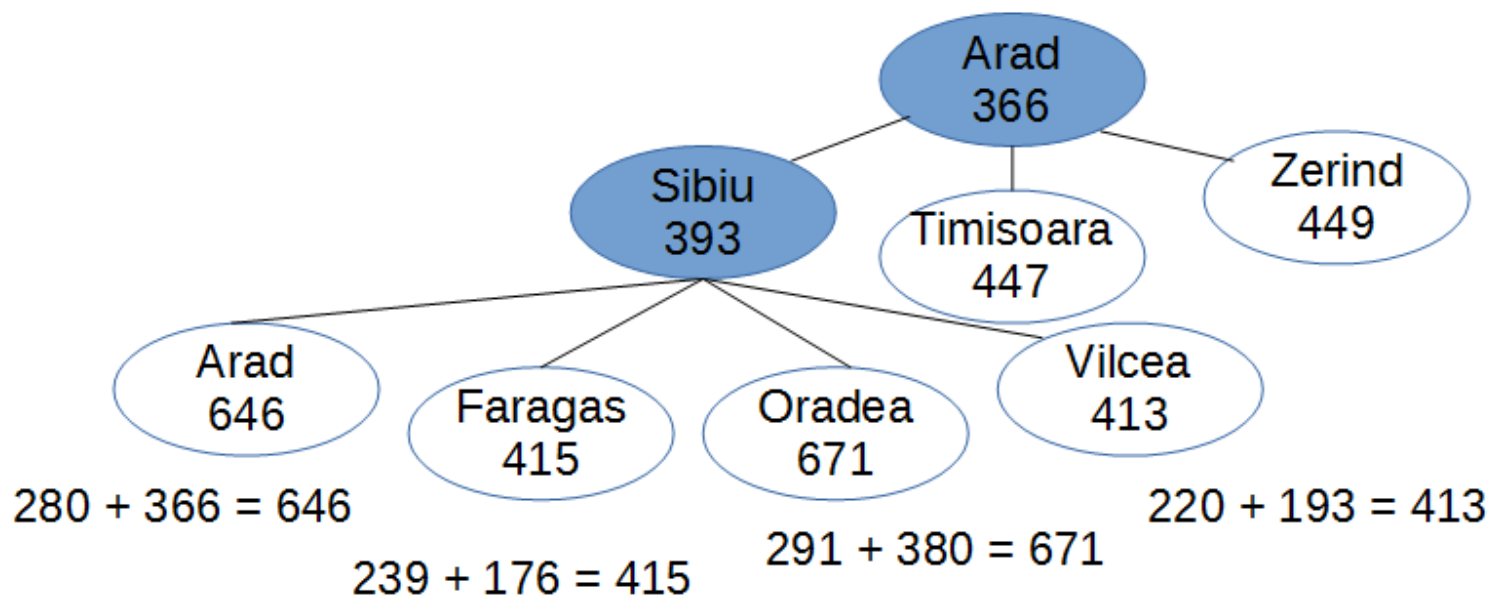
Busca Com Informação - Busca A*

- b) Expansão de Arad



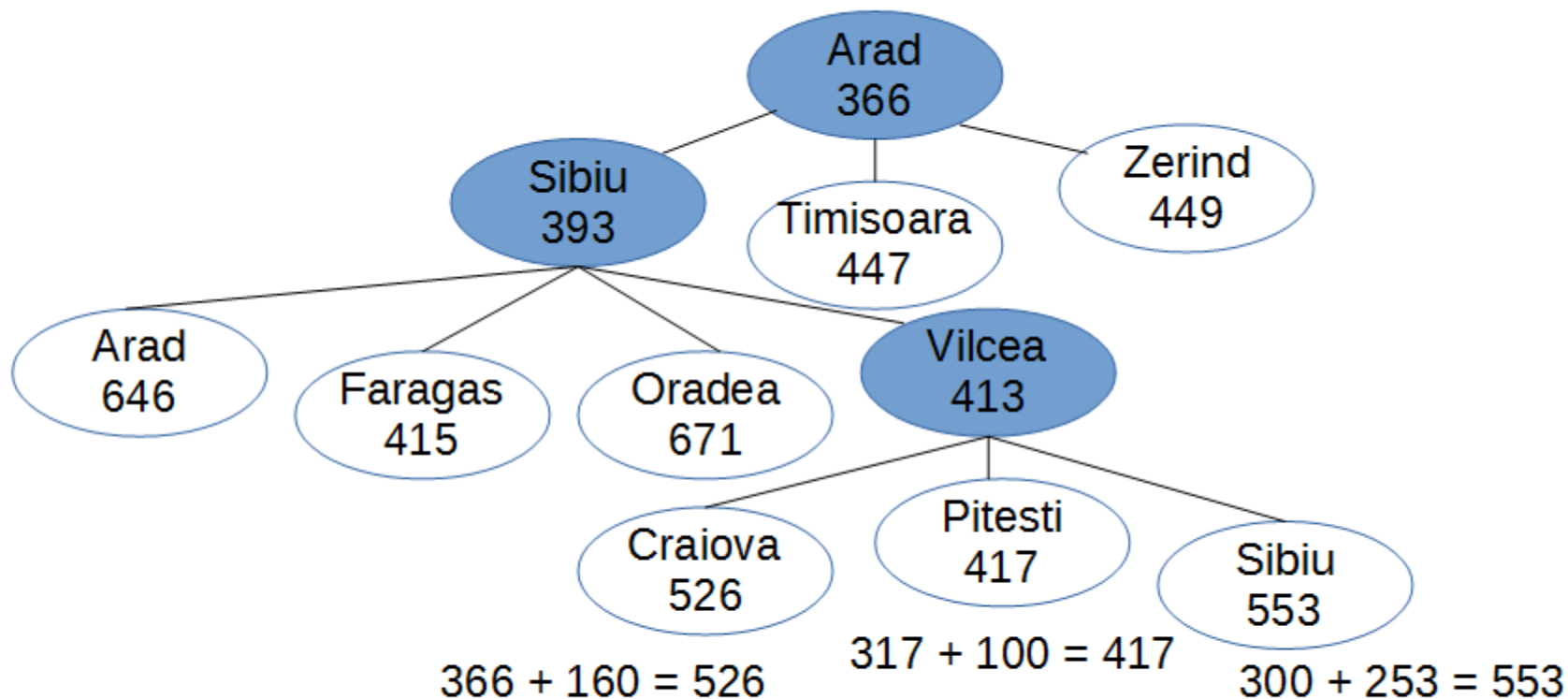
Busca Com Informação - Busca A*

- c) Expansão de Sibiu



Busca Com Informação - Busca A*

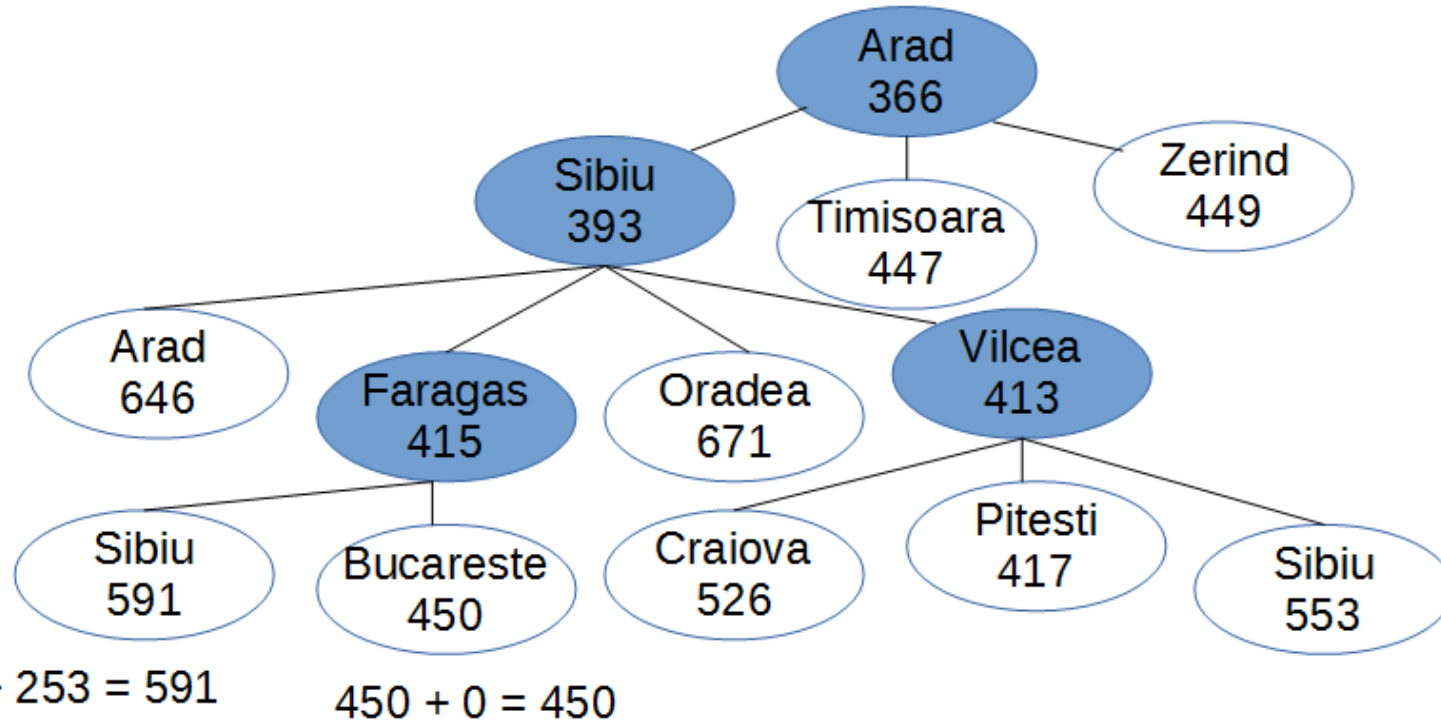
- d) Expansão de Vilcea



Qual nó será expandido agora?

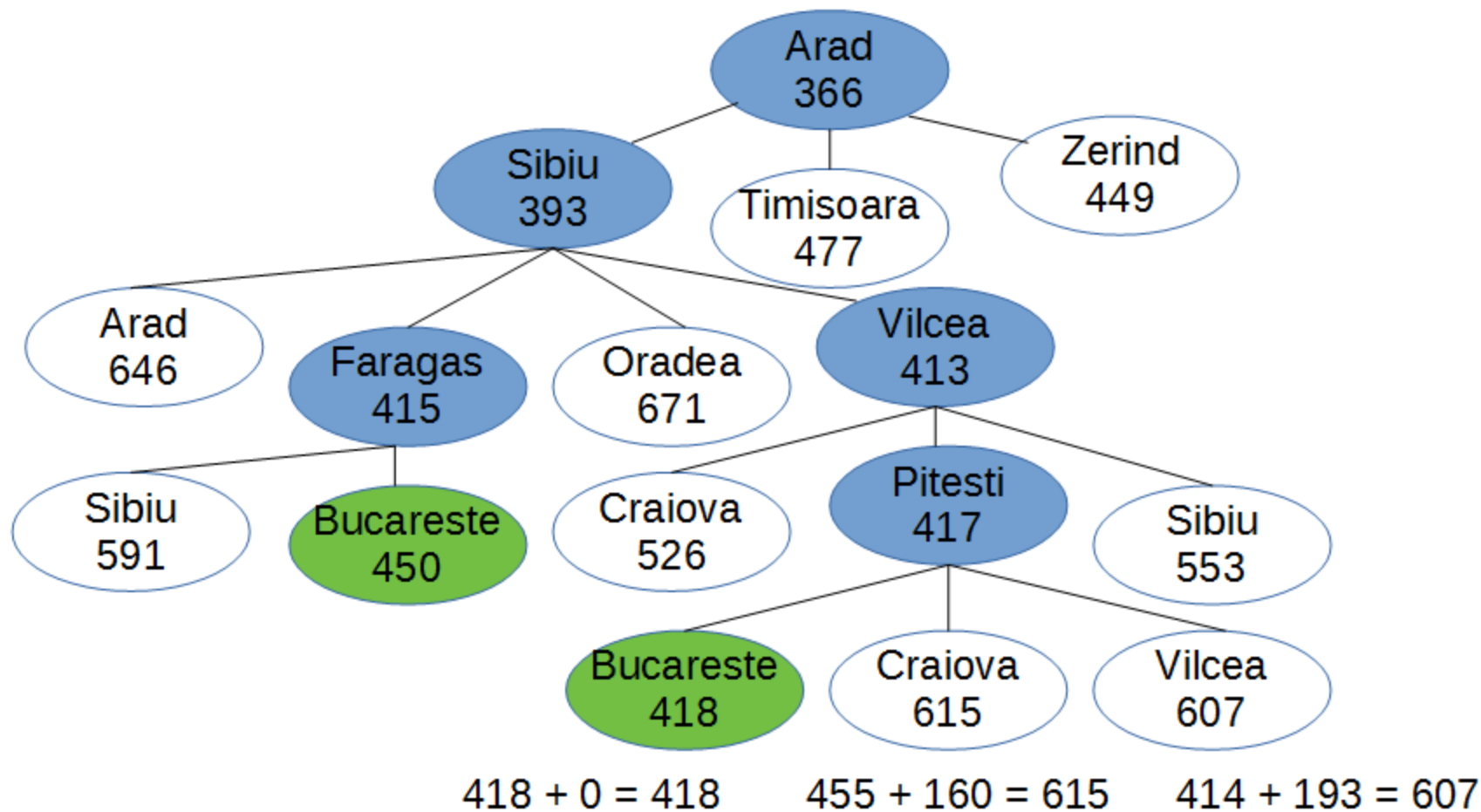
Busca Com Informação - Busca A*

- e) Expansão de Faragas



Busca Com Informação - Busca A*

- f) Expansão de Pitesti



Busca Com Informação - Busca A*

- Se a função heurística $h(n)$ satisfaz algumas condições, a busca A* será completa e ótima
- Duas condições para a otimalidade

1) Admissibilidade: nunca superestima o custo de atingir o objetivo (otimista), ou seja, se para cada nó n , a seguinte condição é satisfeita

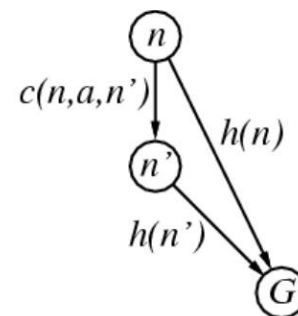
$$h(n) \leq h^*(n)$$

sendo $h^*(n)$ o custo real para alcançar o objetivo a partir do nó n

2) Consistência ou monotonicidade: se para cada nó n , todo sucessor n' gerado pela ação a sobre n obedece a lei da desigualdade triangular (cada um dos lados de um triângulo não pode ser mais longo que a soma dos outros dois lados)

$$h(n) \leq c(n, a, n') + h(n')$$

onde $c(n, a, n')$ é o custo para alcançar n' a partir de n



Busca Com Informação - Busca A*

- Desempenho
- **Completo**: sim, se **b** for finito
- **Otimidade em árvores**: sim, se $h(n)$ for admissível
- **Otimidade em grafos**: sim, se $h(n)$ for consistente
- **Complexidade de tempo**: $O(b^d)$
- **Complexidade de espaço**: $O(b^d)$
- Para mais leituras sobre o algoritmo A*
<https://www.inf.ufsc.br/~alexandre.goncalves.silva/courses/14s2/ine5633/trabalhos/t1/A%20%20Pathfinding%20para%20Iniciantes.pdf>

Busca Com Informação - Busca A*

- **Próxima Aula:**

Buscas com Competição