

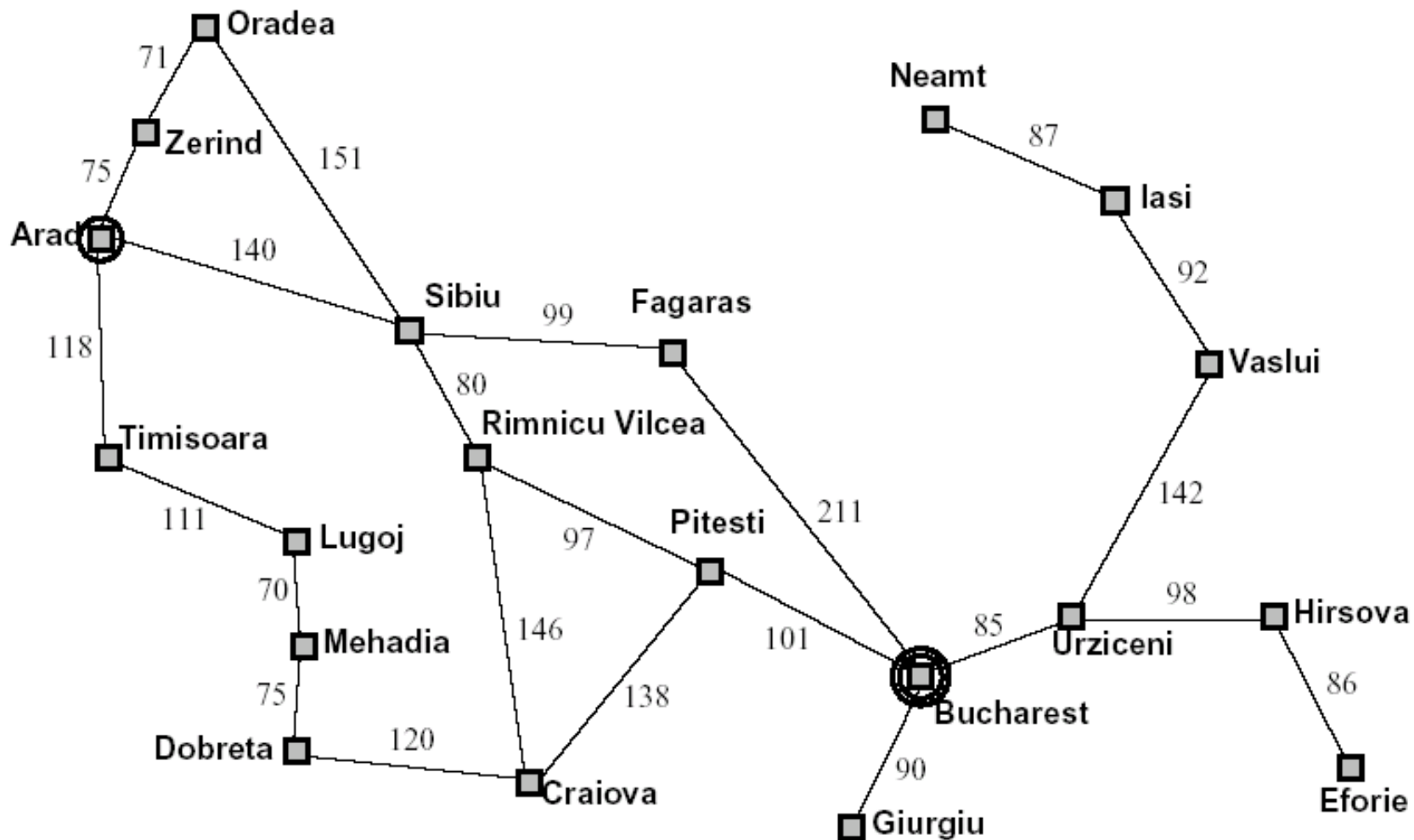
Resolução dos exercícios

Busca em largura

Busca em profundidade

Busca pelo menor curso primeiro

Busca em profundidade



Busca em profundidade

- Estado Inicial: $Em(\text{Arad}) = \text{Arad}$
- Estado Final: $Em(\text{Bucharest}) = \text{Bucharest}$
- Busca:

Começando pelo nó objetivo:

```
[no(Arad, [Arad], 0)]
```

Arad não é o objetivo – Expandindo Arad => Zerind, Sibiu, Timisoara

```
[no(Zerind, [Arad, Zerind], 75),  
no(Sibiu, [Arad, Sibiu], 140),  
no(Timisoara, [Arad, Timisoara], 118)]
```

Zerind não é o objetivo – Expandindo Zerind => Oradea

```
[no(Oradea, [Arad, Zerind, Oradea], 146),  
no(Sibiu, [Arad, Sibiu], 140),  
no(Timisoara, [Arad, Timisoara], 118)]
```

Busca em profundidade

Oradea não é o objetivo – Expandindo Oradea => Sibiu

```
[no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),  
no(Sibiu, [Arad, Sibiu], 140),  
no(Timisoara, [Arad, Timisoara], 118)]
```

Sibiu não é o objetivo – Expandindo Sibiu => Fagaras, Rimnicu Vilcea

```
[no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),  
no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu  
Vilcea], 377),  
no(Sibiu, [Arad, Sibiu], 140),  
no(Timisoara, [Arad, Timisoara], 118)]
```

Busca em profundidade

Fagaras não é o objetivo – Expandindo Fagaras => Bucharest

```
[no(Bucharest, [Arad, Zerind, Oradea, Sibiu, Fagaras, Bucharest],  
607),
```

```
no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu  
Vilcea], 377),
```

```
no(Sibiu, [Arad, Sibiu], 140),
```

```
no(Timisoara, [Arad, Timisoara], 118)]
```

- Bucharest é o objetivo
 - Caminho encontrado [Arad, Zerind, Oradea, Sibiu, Fagaras, Bucharest]
 - Custo=607

Busca em Largura

- Estado Inicial: $Em(Arad) = Arad$
- Estado Final: $Em(Bucharest) = Bucharest$
- Busca:

Começando pelo nó objetivo:

```
[no(Arad, [Arad], 0)]
```

Arad não é o objetivo – Expandindo Arad => Zerind, Sibiu, Timisoara

```
[no(Zerind, [Arad, Zerind], 75),  
no(Sibiu, [Arad, Sibiu], 140),  
no(Timisoara, [Arad, Timisoara], 118)]
```

Zerind não é o objetivo – Expandindo Zerind => Oradea

```
[no(Sibiu, [Arad, Sibiu], 140),  
no(Timisoara, [Arad, Timisoara], 118),  
no(Oradea, [Arad, Zerind, Oradea], 146)]
```

Busca em Largura

Sibiu não é o objetivo – Expandindo Sibiu => Fagaras, Rimnicu Vilcea

```
[no(Timisoara, [Arad, Timisoara], 118),  
 no(Oradea, [Arad, Zerind, Oradea], 146),  
 no(Fagaras, [Arad, Sibiu, Fagaras], 239),  
 no(Rimnicu Vilcea, [Arad, Sibiu, Rimnicu Vilcea], 320)]
```

Timisoara não é o objetivo – Expandindo Timisoara => Lugoj

```
[no(Oradea, [Arad, Zerind, Oradea], 146),  
 no(Fagaras, [Arad, Sibiu, Fagaras], 239),  
 no(Rimnicu Vilcea, [Arad, Sibiu, Rimnicu Vilcea], 320),  
 no(Lugoj, [Arad, Timisoara, Lugoj], 229)]
```

Busca em Largura

Oradea não é o objetivo – Expandindo Oradea => Sibiu

```
[no(Fagaras, [Arad, Sibiu, Fagaras], 239),  
 no(Rimnicu Vilcea, [Arad, Sibiu, Rimnicu Vilcea], 320),  
 no(Lugoj, [Arad, Timisoara, Lugoj], 229),  
 no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297)]
```

Fagaras não é o objetivo – Expandindo Fagaras => Bucharest

```
[no(Rimnicu Vilcea, [Arad, Sibiu, Rimnicu Vilcea], 320),  
 no(Lugoj, [Arad, Timisoara, Lugoj], 229),  
 no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),  
 no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450)]
```


Busca em Largura

Rimnicu Vilcea não é o objetivo – Expandindo Rimnicu Vilcea=>Pitest, Craiova

```
[no(Lugoj, [Arad, Timisoara, Lugoj], 229),  
no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),  
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),  
no(Pitest, [Arad, Sibiu, Rimnicu Vilcea, Pitest], 417),  
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 466)]
```

Lugoj não é o objetivo – Expandindo Lugoj=>Mehadia

```
[no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),  
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),  
no(Pitest, [Arad, Sibiu, Rimnicu Vilcea, Pitest], 417),  
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 466),  
no(Mehadia, [Arad, Timisoara, Lugoj, Mehadia], 299)]
```

Busca em Largura

Sibiu não é o objetivo – Expandindo Sibiu=>Fagaras, Rimnicu Vilcea

```
[no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),  
no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Pitesti], 417),  
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 466),  
no(Mehadia, [Arad, Timisoara, Lugoj, Mehadia], 299),  
no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),  
no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu  
Vilcea], 377)]
```

- Bucharest é o objetivo
 - Caminho encontrado: [Arad, Sibiu, Fagaras, Bucharest]
 - Custo=450

Busca pelo menor custo primeiro

- Estado Inicial: $Em(Arad) = Arad$
- Estado Final: $Em(Bucharest) = Bucharest$
- Busca:

Começando pelo nó objetivo:

```
[no(Arad, [Arad], 0)]
```

Arad não é o objetivo – Expandindo Arad => Zerind, Sibiu, Timisoara

```
[no(Zerind, [Arad, Zerind], 75),  
no(Timisoara, [Arad, Timisoara], 118),  
no(Sibiu, [Arad, Sibiu], 140)]
```

Zerind não é o objetivo – Expandindo Zerind => Oradea

```
[no(Timisoara, [Arad, Timisoara], 118),  
no(Sibiu, [Arad, Sibiu], 140),  
no(Oradea, [Arad, Zerind, Oradea], 146)]
```

Busca pelo menor custo primeiro

Timisoara não é o objetivo – Expandindo Timisoara=>Lugoj

```
[no(Sibiu, [Arad, Sibiu], 140),  
no(Oradea, [Arad, Zerind, Oradea], 146),  
no(Lugoj, [Arad, Timisoara, Lugoj], 229)]
```

Sibiu não é o objetivo – Expandindo Sibiu=>Fagaras, Rimnicu Vilcea

```
[no(Oradea, [Arad, Zerind, Oradea], 146),  
no(Rimnicu Vilcea, [Arad, Sibiu, Rimnicu Vilcea], 220),  
no(Lugoj, [Arad, Timisoara, Lugoj], 229),  
no(Fagaras, [Arad, Sibiu, Fagaras], 239)]
```

Oradea não é o objetivo – Expandindo Oradea=>Sibiu

```
[no(Rimnicu Vilcea, [Arad, Sibiu, Rimnicu Vilcea], 220),  
no(Lugoj, [Arad, Timisoara, Lugoj], 229),  
no(Fagaras, [Arad, Sibiu, Fagaras], 239),  
no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297)]
```

Busca pelo menor custo primeiro

Rimnicu Vilcea não é o objetivo – Expandindo Rimnicu Vilcea=>Pitesti, Craiova

```
[no(Lugoj, [Arad, Timisoara, Lugoj], 229),  
no(Fagaras, [Arad, Sibiu, Fagaras], 239),  
no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),  
no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Pitesti], 317),  
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 366)]
```

Lugoj não é o objetivo – Expandindo Lugoj=>Mehadia

```
[no(Fagaras, [Arad, Sibiu, Fagaras], 239),  
no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),  
no(Mehadia, [Arad, Timisoara, Lugoj, Mehadia], 299),  
no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Pitesti], 317),  
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 366)]
```

Busca pelo menor custo primeiro

Fagaras não é o objetivo – Expandindo Fagaras=>Bucharest

[no(Sibiu, [Arad, Zerind, Oradea, Sibiu], 297),
no(Mehadia, [Arad, Timisoara, Lugoj, Mehadia], 299),
no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Pitesti], 317),
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 366),
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450)]

Sibiu não é o objetivo – Expandindo Sibiu=>Fagaras, Rimnicu Vilcea

[no(Mehadia, [Arad, Timisoara, Lugoj, Mehadia], 299),
no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Pitesti], 317),
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 366),
no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu
Vilcea], 377),
no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450)]

Busca pelo menor custo primeiro

Mehadia não é o objetivo – Expandindo Mehadia=>Dobreta

[no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Pitesti], 317),
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 366),
no(Dobreta, [Arad, Timisoara, Lugoj, Mehadia, Dobreta], 374),
no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu
Vilcea], 377),
no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450)]

Pitesti não é o objetivo – Expandindo Pitesti=>Bucharest, Craiova

[no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Craiova], 366),
no(Dobreta, [Arad, Timisoara, Lugoj, Mehadia, Dobreta], 374),
no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu
Vilcea], 377),
no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),
no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitesti, Bucharest],
418),
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitesti, Craiova],
455)]

Busca pelo menor custo primeiro

Craiova não é o objetivo – Expandindo Craiova=>Pitesti, Dobreta

[no(Dobreta, [Arad, Timisoara, Lugoj, Mehadia, Dobreta], 374) ,

*no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu
Vilcea], 377) ,*

no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396) ,

no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Pitesti], 404) ,

*no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitesti, Bucharest],
418) ,*

no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450) ,

*no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitesti, Craiova],
455) ,*

*no(Dobreta, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Dobreta],
486)]*

Busca pelo menor custo primeiro

Dobreta não é o objetivo – Expandindo Dobreta=>Craiova

`[no(Rimnicu Vilcea, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea], 377),`

`no(Craiova, [Arad, Timisoara, Lugoj, Mehadia, Dobreta, Craiova], 394),`

`no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),`

`no(Pitest, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Pitest], 404),`

`no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Bucharest], 418),`

`no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),`

`no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Craiova], 455),`

`no(Dobreta, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Dobreta], 486)]`

Busca pelo menor custo primeiro

Rimnicu Vilcea não é o objetivo – Expandindo Rimnicu Vilcea=>Pitest, Craiova

[no(Craiova, [Arad, Timisoara, Lugoj, Mehadia, Dobreta, Craiova], 394),

no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),

no(Pitest, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Pitest], 404),

no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Bucharest], 418),

no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),

no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Craiova], 455),

no(Pitest, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea, Pitest], 474),

no(Dobreta, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Dobreta], 486),

no(Craiova, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea, Craiova], 523)]

Busca pelo menor custo primeiro

Craiova não é o objetivo – Expandindo Craiova=>Rimnicu Vilcea, Pitest

[no(Fagaras, [Arad, Zerind, Oradea, Sibiu, Fagaras], 396),
no(Pitest, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Pitest], 404),
no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Bucharest],
418),
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Craiova],
455),
no(Pitest, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea,
Pitest], 474),
no(Dobreta, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Dobreta],
486),
no(Craiova, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea,
Craiova], 523),
no(Pitest, [Arad, Timisoara, Lugoj, Mehadia, Dobreta, Craiova,
Pitest], 532),
no(Rimnicu Vilcea, [Arad, Timisoara, Lugoj, Mehadia, Dobreta,
Craiova, Rimnicu Vilcea], 540)]

Busca pelo menor custo primeiro

Fagaras não é o objetivo – Expandindo Fagaras=>Bucharest

[no(Pitesti, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Pitesti], 404),
no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitesti, Bucharest],
418),
no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450),
no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitesti, Craiova],
455),
no(Pitesti, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea,
Pitesti], 474),
no(Dobreta, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Dobreta],
486),
no(Craiova, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea,
Craiova], 523),
no(Pitesti, [Arad, Timisoara, Lugoj, Mehadia, Dobreta, Craiova,
Pitesti], 532),
no(Rimnicu Vilcea, [Arad, Timisoara, Lugoj, Mehadia, Dobreta,
Craiova, Rimnicu Vilcea], 540),
no(Bucharest, [Arad, Zerind, Oradea, Sibiu, Fagaras, Bucharest],
607)]

Busca pelo menor custo primeiro

Pitest não é o objetivo – Expandindo Pitest=>Bucharest

[no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Bucharest], 418) ,

no(Bucharest, [Arad, Sibiu, Fagaras, Bucharest], 450) ,

no(Craiova, [Arad, Sibiu, Rimnicu Vilcea, Pitest, Craiova], 455) ,

no(Pitest, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea, Pitest], 474) ,

no(Dobreta, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Dobreta], 486) ,

no(Bucharest, [Arad, Sibiu, Rimnicu Vilcea, Craiova, Pitest, Bucharest], 505) ,

no(Craiova, [Arad, Zerind, Oradea, Sibiu, Rimnicu Vilcea, Craiova], 523) ,

no(Pitest, [Arad, Timisoara, Lugoj, Mehadia, Dobreta, Craiova, Pitest], 532) ,

no(Rimnicu Vilcea, [Arad, Timisoara, Lugoj, Mehadia, Dobreta, Craiova, Rimnicu Vilcea], 540) ,

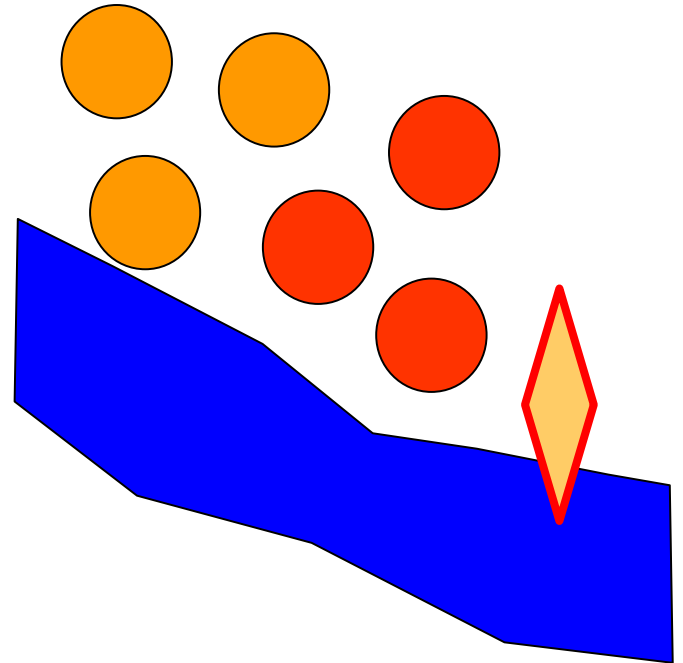
no(Bucharest, [Arad, Zerind, Oradea, Sibiu, Fagaras, Bucharest], 607)]

Busca pelo menor custo primeiro

- Bucharest é o objetivo
 - Caminho encontrado: [Arad, Sibiu, Rimnicu Vilcea, Pitest, Bucharest]
 - Custo: 418
- Busca em Largura
 - Caminho encontrado: [Arad, Sibiu, Fagaras, Bucharest]
 - Custo=450
- Busca em Profundidade
 - Caminho encontrado [Arad, Zerind, Oradea, Sibiu, Fagaras, Bucharest]
 - Custo=607

Exercício: Resolva o problema dos Missionários e Canibais

- Represente os estados da forma: $\langle M,C,B \rangle$
- Função sucessor:
 - Levar 1 missionário e 1 canibal,
 - Levar 2 missionários,
 - Levar 2 canibais,
 - Levar 1 missionário,
 - Levar 1 canibal
- Resolva por busca em largura e por busca em profundidade



Busca em profundidade

- Estado Inicial: $\langle 3,3,1 \rangle$
- Estado Final: $\langle 0,0,0 \rangle$
- Busca:

Começando pelo nó objetivo:

[no($\langle 331 \rangle$, [$\langle 331 \rangle$])]

$\langle 331 \rangle$ não é o objetivo. Expandindo $\langle 331 \rangle \Rightarrow \langle 220 \rangle, \langle 310 \rangle, \langle 320 \rangle$

[no($\langle 220 \rangle$, [$\langle 331 \rangle, \langle 220 \rangle$]), no($\langle 310 \rangle$, [$\langle 331 \rangle, \langle 310 \rangle$]), no($\langle 320 \rangle$, [$\langle 331 \rangle, \langle 320 \rangle$])]

$\langle 220 \rangle$ não é o objetivo. Expandindo $\langle 220 \rangle \Rightarrow \langle 331 \rangle^*, \langle 321 \rangle$

[no($\langle 321 \rangle$, [$\langle 331 \rangle, \langle 220 \rangle, \langle 321 \rangle$]), no($\langle 310 \rangle$, [$\langle 331 \rangle, \langle 310 \rangle$]), no($\langle 320 \rangle$, [$\langle 331 \rangle, \langle 320 \rangle$])]

Busca em profundidade

<321> não é o objetivo. Expandindo <321> => <300>, <220>*, <310>

```
[no (<300>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 310 \rangle$ ]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 310 \rangle$ ]), no (<320>, [ $\langle 331 \rangle$ ,  $\langle 320 \rangle$ ])]
```

<300> não é o objetivo. Expandindo <300> => <321>*, <311>

```
[no (<311>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ,  $\langle 311 \rangle$ ]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 310 \rangle$ ]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 310 \rangle$ ]), no (<320>, [ $\langle 331 \rangle$ ,  $\langle 320 \rangle$ ])]
```

<311> não é o objetivo. Expandindo <311> => <110>, <330>*

```
[no (<110>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ,  $\langle 311 \rangle$ ,  $\langle 110 \rangle$ ]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 310 \rangle$ ]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 310 \rangle$ ]), no (<320>, [ $\langle 331 \rangle$ ,  $\langle 320 \rangle$ ])]
```

Busca em profundidade

<110> não é o objetivo. Expandindo <110> => <311>*, <221>

```
[no (<221>, [<331>, <220>, <321>, <300>, <311>, <110>, <221>]),  
no (<310>, [<331>, <220>, <321>, <310>]),  
no (<310>, [<331>, <310>]), no (<320>, [<331>, <320>])]
```

<221> não é o objetivo. Expandindo <221> => <110>*, <020>

```
[no (<020>, [<331>, <220>, <321>, <300>, <311>, <110>, <221>,  
<020>]),  
no (<310>, [<331>, <220>, <321>, <310>]),  
no (<310>, [<331>, <310>]), no (<320>, [<331>, <320>])]
```

<020> não é o objetivo. Expandindo <020> => <221>*, <031>

```
[no (<030>, [<331>, <220>, <321>, <300>, <311>, <110>, <221>,  
<020>, <031>]),  
no (<310>, [<331>, <220>, <321>, <310>]),  
no (<310>, [<331>, <310>]), no (<320>, [<331>, <320>])]
```

Busca em profundidade

<031> não é o objetivo. Expandindo <031> => <010>, <020>*

```
[no (<010>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ,  $\langle 311 \rangle$ ,  $\langle 110 \rangle$ ,  $\langle 221 \rangle$ ,  
  <020>, <031>, <010>]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ , <310>]),  
no (<310>, [ $\langle 331 \rangle$ , <310>]), no (<320>, [ $\langle 331 \rangle$ , <320>])]
```

<010> não é o objetivo. Expandindo <010> => <031>*, <021>

```
[no (<021>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ,  $\langle 311 \rangle$ ,  $\langle 110 \rangle$ ,  $\langle 221 \rangle$ ,  
  <020>, <031>, <010>, <021>]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ , <310>]),  
no (<310>, [ $\langle 331 \rangle$ , <310>]), no (<320>, [ $\langle 331 \rangle$ , <320>])]
```

<021> não é o objetivo. Expandindo <021> => <000>, <010>*

```
[no (<000>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ,  $\langle 311 \rangle$ ,  $\langle 110 \rangle$ ,  $\langle 221 \rangle$ ,  
  <020>, <031>, <010>, <021>, <000>]),  
no (<310>, [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ , <310>]),  
no (<310>, [ $\langle 331 \rangle$ , <310>]), no (<320>, [ $\langle 331 \rangle$ , <320>])]
```

Busca em profundidade

$\langle 021 \rangle$ não é o objetivo. Expandindo $\langle 021 \rangle \Rightarrow \langle 000 \rangle, \langle 010 \rangle^*$

```
[no( $\langle 000 \rangle$ , [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 300 \rangle$ ,  $\langle 311 \rangle$ ,  $\langle 110 \rangle$ ,  $\langle 221 \rangle$ ,  
   $\langle 020 \rangle$ ,  $\langle 031 \rangle$ ,  $\langle 010 \rangle$ ,  $\langle 021 \rangle$ ,  $\langle 000 \rangle$ ]),  
no( $\langle 310 \rangle$ , [ $\langle 331 \rangle$ ,  $\langle 220 \rangle$ ,  $\langle 321 \rangle$ ,  $\langle 310 \rangle$ ]),  
no( $\langle 310 \rangle$ , [ $\langle 331 \rangle$ ,  $\langle 310 \rangle$ ]), no( $\langle 320 \rangle$ , [ $\langle 331 \rangle$ ,  $\langle 320 \rangle$ ])]
```

- $\langle 000 \rangle$ é o objetivo
 - Caminho encontrado: [$\langle 331 \rangle$, $\langle 220 \rangle$, $\langle 321 \rangle$, $\langle 300 \rangle$, $\langle 311 \rangle$, $\langle 110 \rangle$,
 $\langle 221 \rangle$, $\langle 020 \rangle$, $\langle 031 \rangle$, $\langle 010 \rangle$, $\langle 021 \rangle$, $\langle 000 \rangle$]

Busca em Largura

- Estado Inicial: $\langle 3,3,1 \rangle$
- Estado Final: $\langle 0,0,0 \rangle$
- Busca:

Começando pelo nó objetivo:

`[no (<331>, [<331>])]`

$\langle 331 \rangle$ não é o objetivo. Expandindo $\langle 331 \rangle \Rightarrow \langle 220 \rangle, \langle 310 \rangle, \langle 320 \rangle$

`[no (<220>, [<331>, <220>]), no (<310>, [<331>, <310>]), no (<320>, [<331>, <320>])]`

$\langle 220 \rangle$ não é o objetivo. Expandindo $\langle 220 \rangle \Rightarrow \langle 331 \rangle^*, \langle 321 \rangle$

`[no (<310>, [<331>, <310>]), no (<320>, [<331>, <320>]), no (<321>, [<331>, <220>, <321>])]`

A continuação fica como exercício...