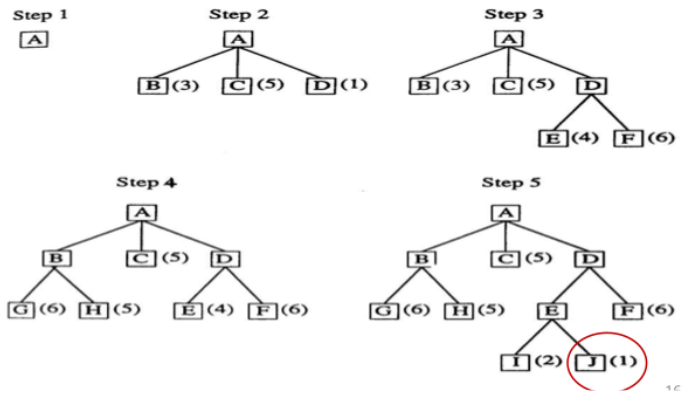
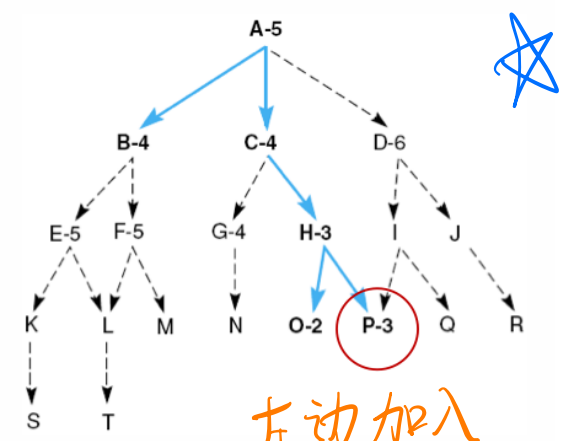


并不是只选择当前最小的，而是与其它的一起考量.



★注意：open需要 sort !!!

左边加入

open

A 5

B4 C4 D6

C4 E5 F5 D6

H3 G4 E5 F5 D6

O2 P3 H3 G4 E5 F5 D6

P3 H3 G4 E5 F5 D6

↑

Goal

左边加入

closed

[ ]

A5

B4 A5

C4 B4 A5

H3 C4 B4 A5

O2 H3 C4 B4 A5

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

其中 $g(n)$: path-cost function 可以理解为过去的值.

$h(n)$: heuristic function 可以理解为未来的值.

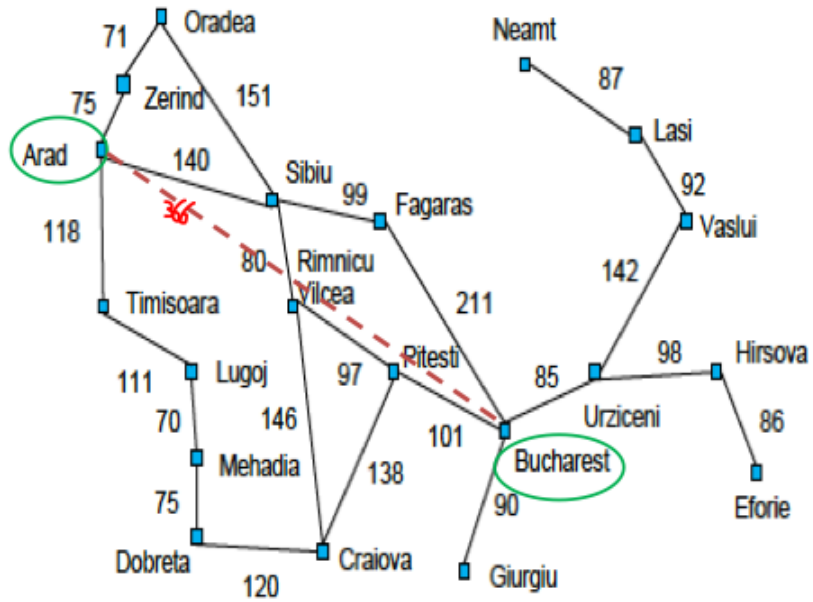
而 greedy Best first 中 $g(n)=0$ 即不看过去.

Example: (greedy) Best-First** to Find a Path from **Arad** to **Bucharest

$h(n)$: evaluation function =estimate of cost from n to the closest goal

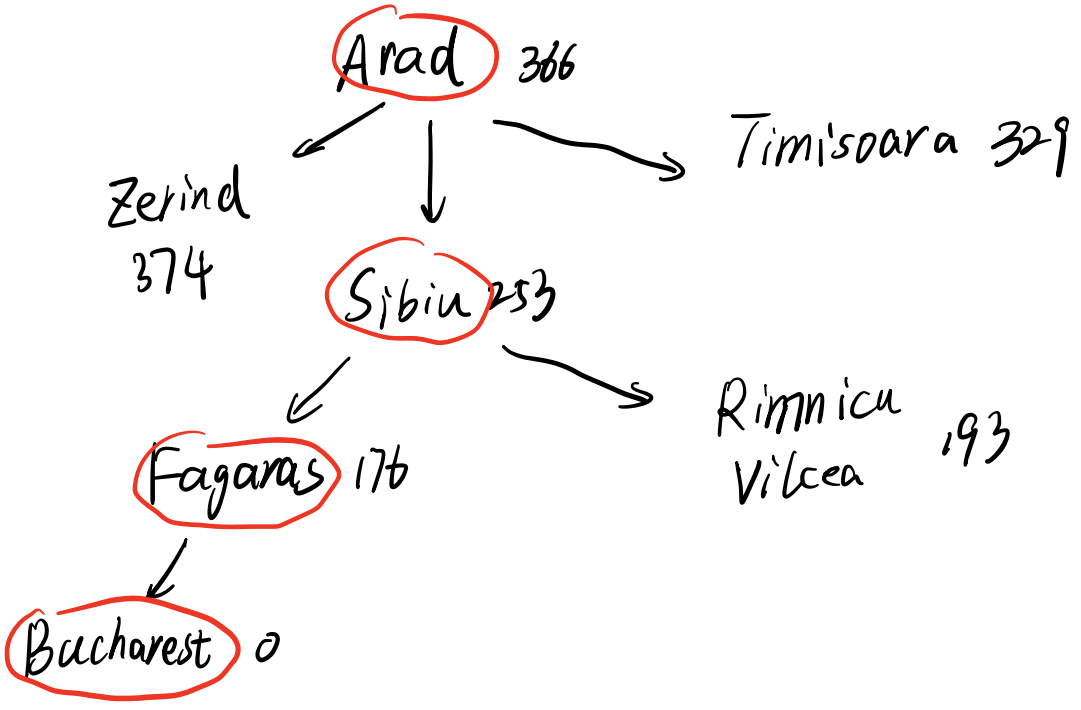
$h_{SLD}(n)$ = straight-line distance from n to **Bucharest**

$f(n) = h_{SLD}(n)$ $g(n)=0$



$h(n)$ SLD from a to b

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



相当于

open

Arad 366

Sibiu 253 Timisoara 329 Zerind 374

F 176 R 193 T 329 Z 374

B 0 R 193 T 329 Z 374

↑

Goal

closed

[ ]

Arad 366

Sibiu 253 Arad 366

F 176 ~