

Pengantar Kecerdasan Buatan

INFORMED SEARCH



Informed Search

- Uninformed searches
 - easy
 - but very inefficient in most cases of huge search tree
- Informed searches
 - uses problem-specific information to reduce the search tree into a small one
 - resolve time and memory complexities

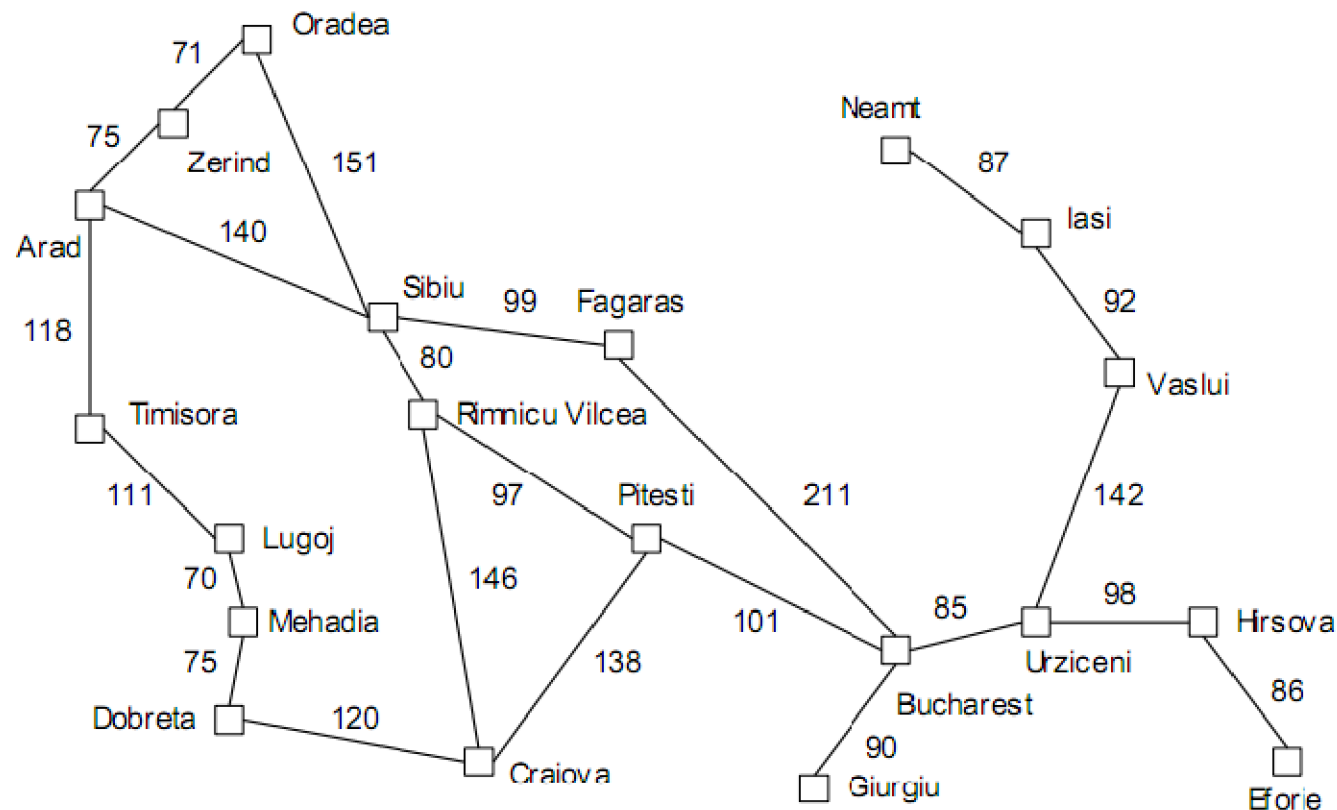
Informed Search

- Greedy best-first search
- A* search
- Uniform Cost Search

Another method :

- Heuristics
- Local search algorithms
- Hill-climbing search
- Simulated annealing search
- Local beam search
- Genetic algorithms

Romania with Straight-Line dist.



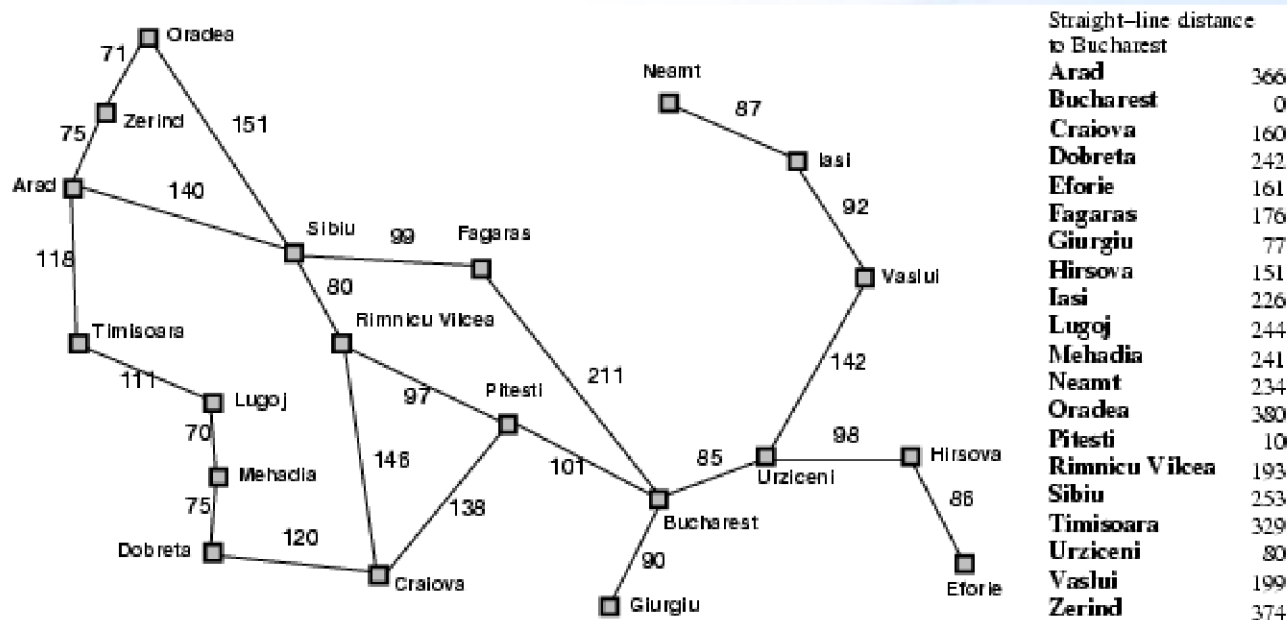
Straight-line distance
to Bucharest

Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	226
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

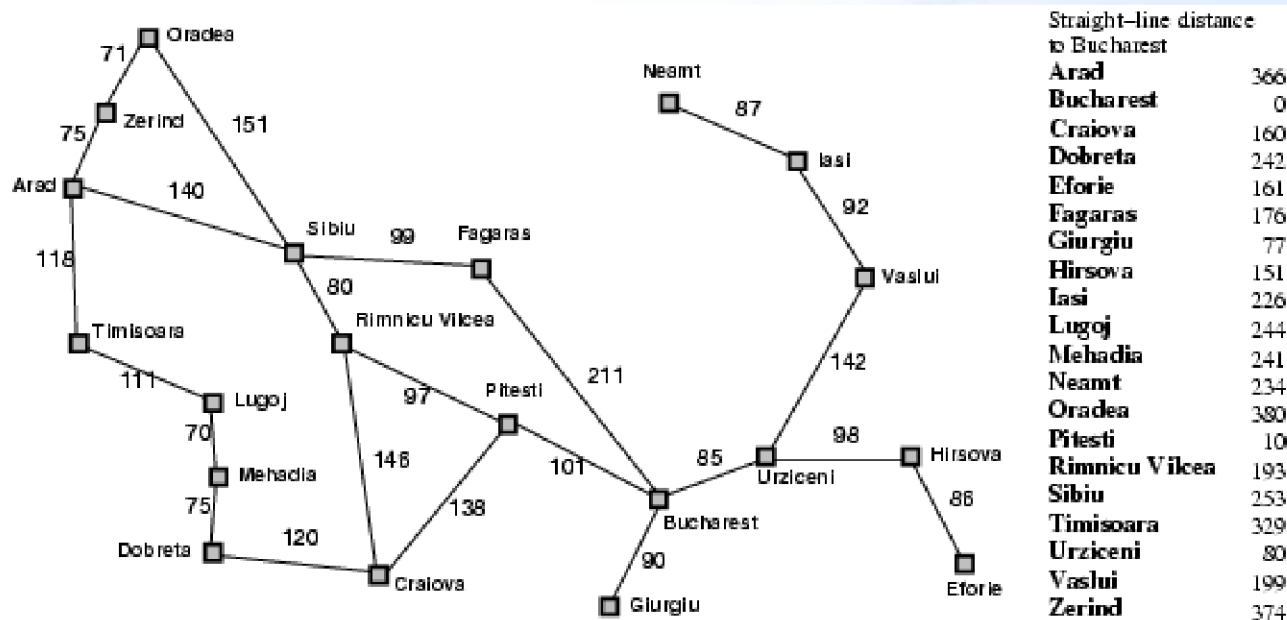
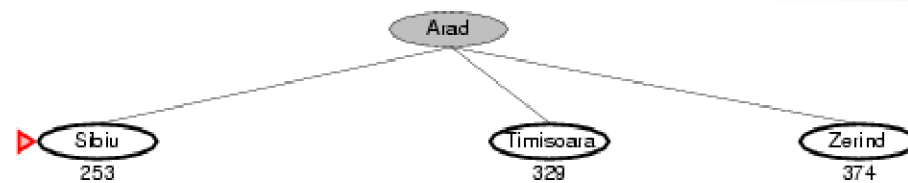
Greedy best-first search

- $f(n)$ = estimate of cost from n to *goal*
- e.g., $f_{SLD}(n)$ = straight-line distance from n to Bucharest
- Greedy best-first search expands the node that **appears** to be closest to goal.

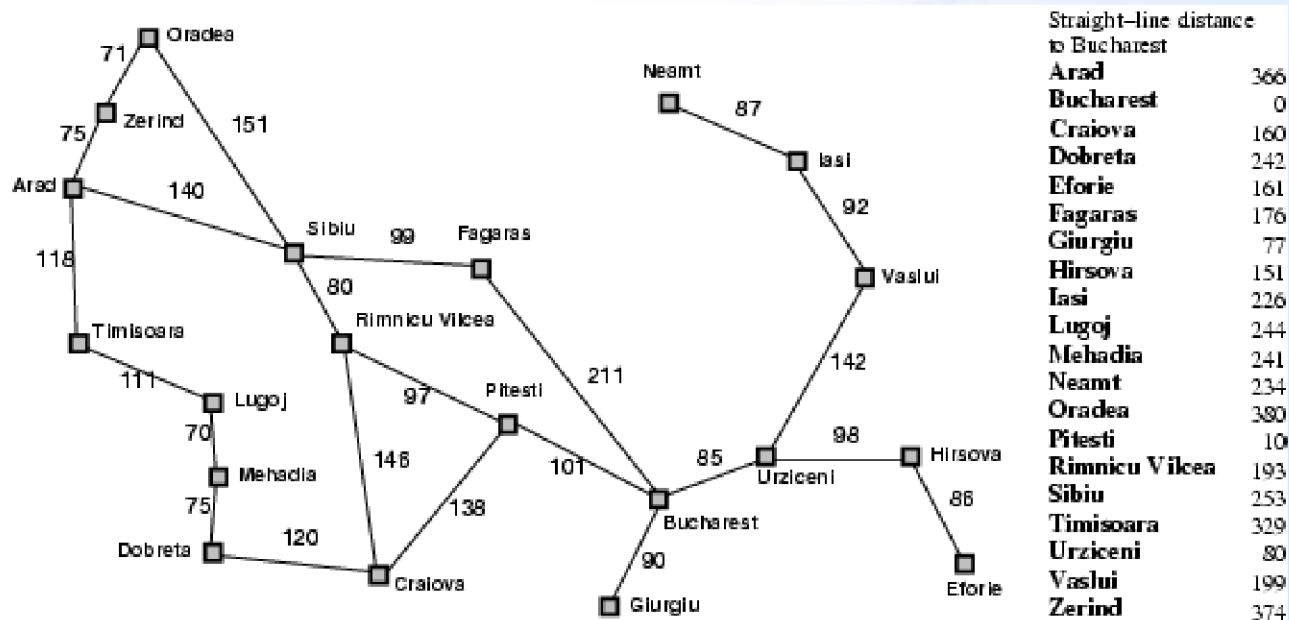
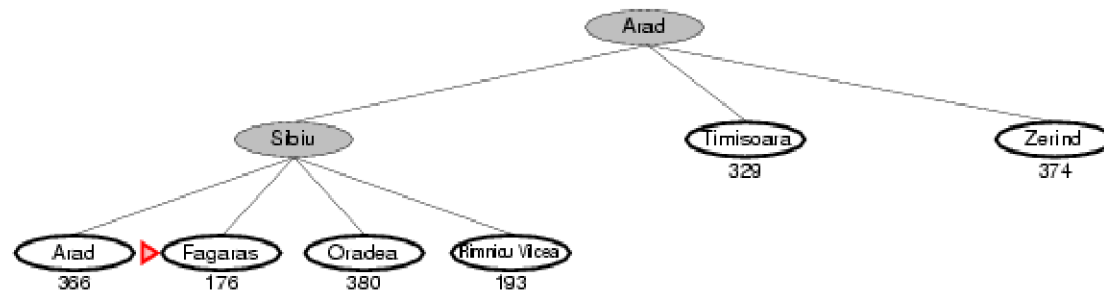
Greedy best-first search example



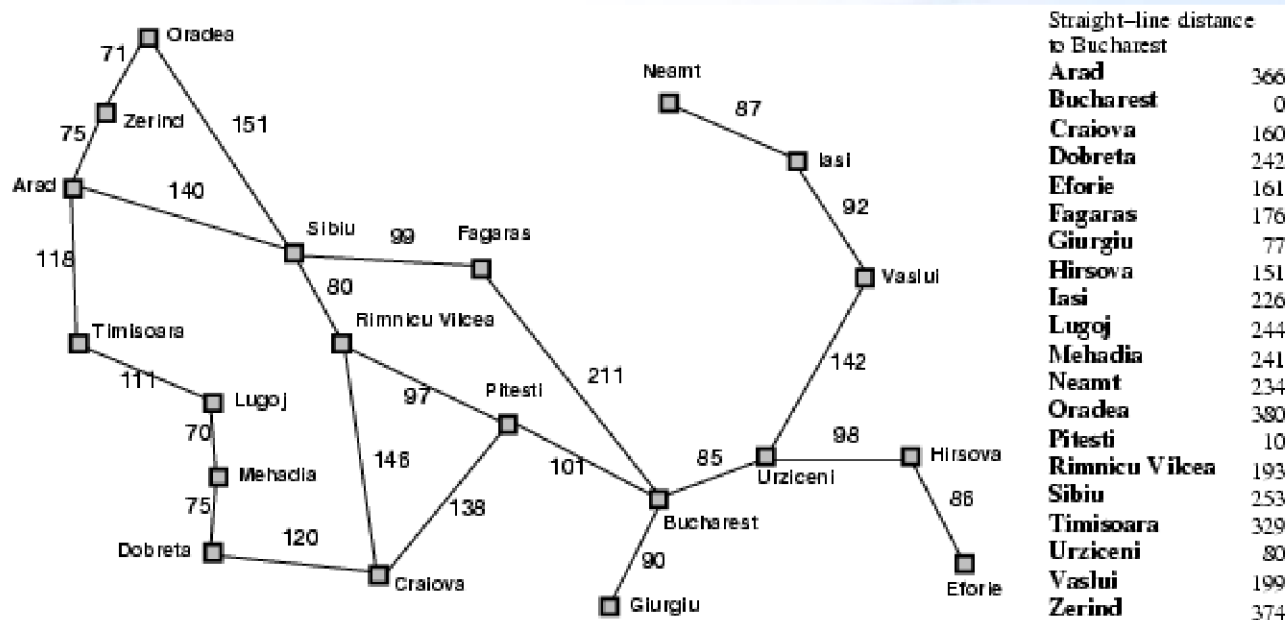
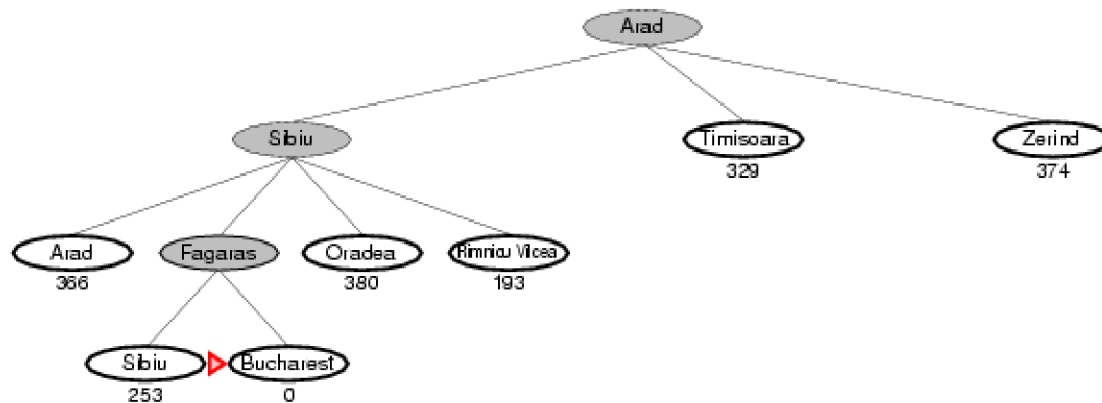
Greedy best-first search example



Greedy best-first search example



Greedy best-first search example



Properties of greedy best-first search

- Complete? No – can get stuck in loops.
- Time? $O(b^m)$, but a good heuristic can give dramatic improvement
- Space? $O(b^m)$ - keeps all nodes in memory
- Optimal? No

e.g. Arad → Sibiu → Rimnicu

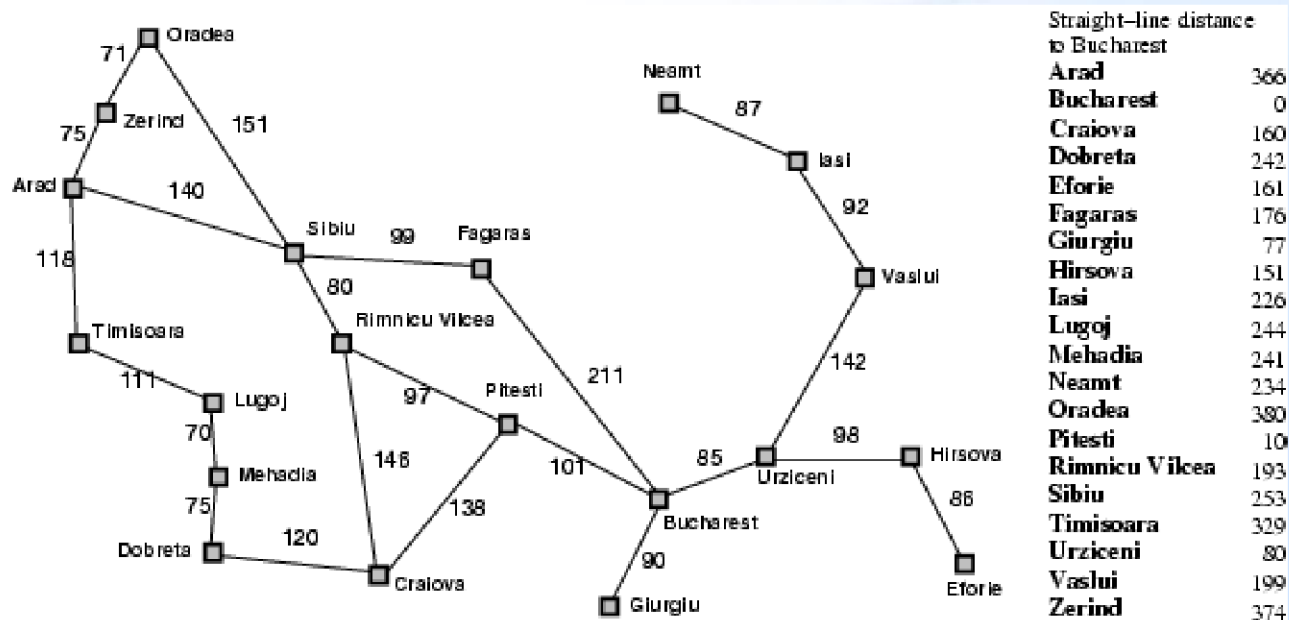
Virea → Pitesti → Bucharest is shorter!

A* search

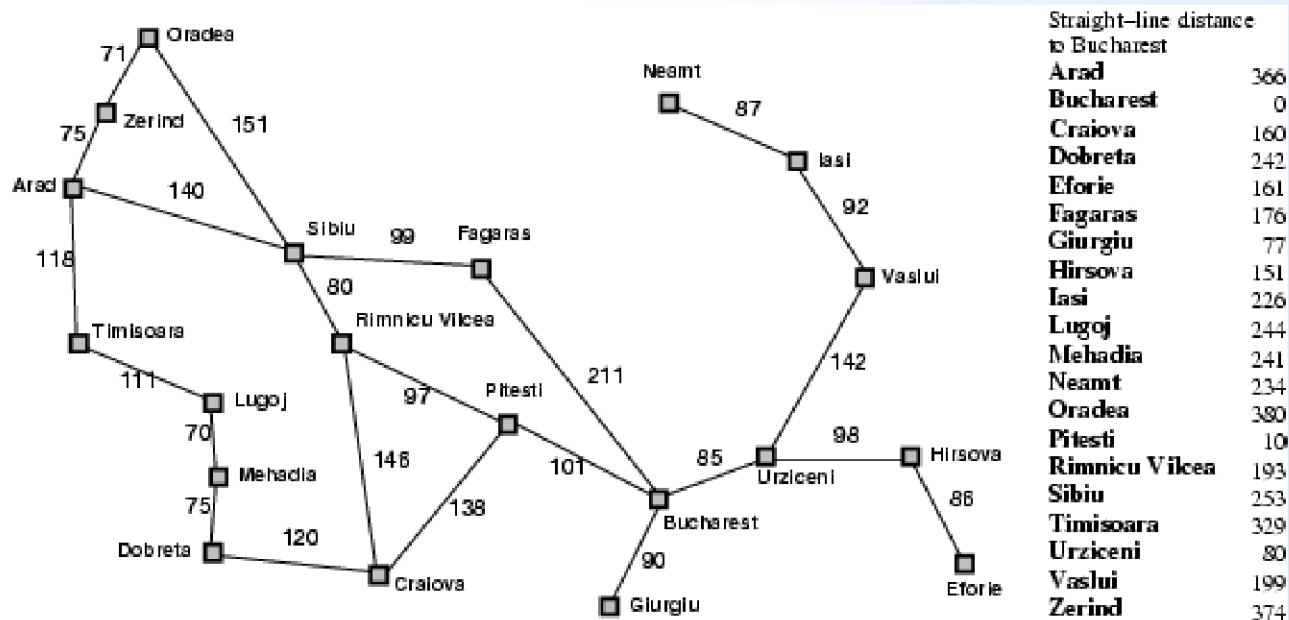
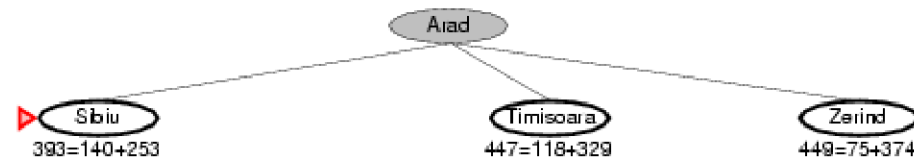
- Idea: avoid expanding paths that are already expensive
- Evaluation function $f(n) = g(n) + h(n)$
- $g(n)$ = cost so far to reach n
- $h(n)$ = estimated cost from n to goal
- $f(n)$ = estimated total cost of path through n to goal
- Best First search has $f(n) = h(n)$

A* search example

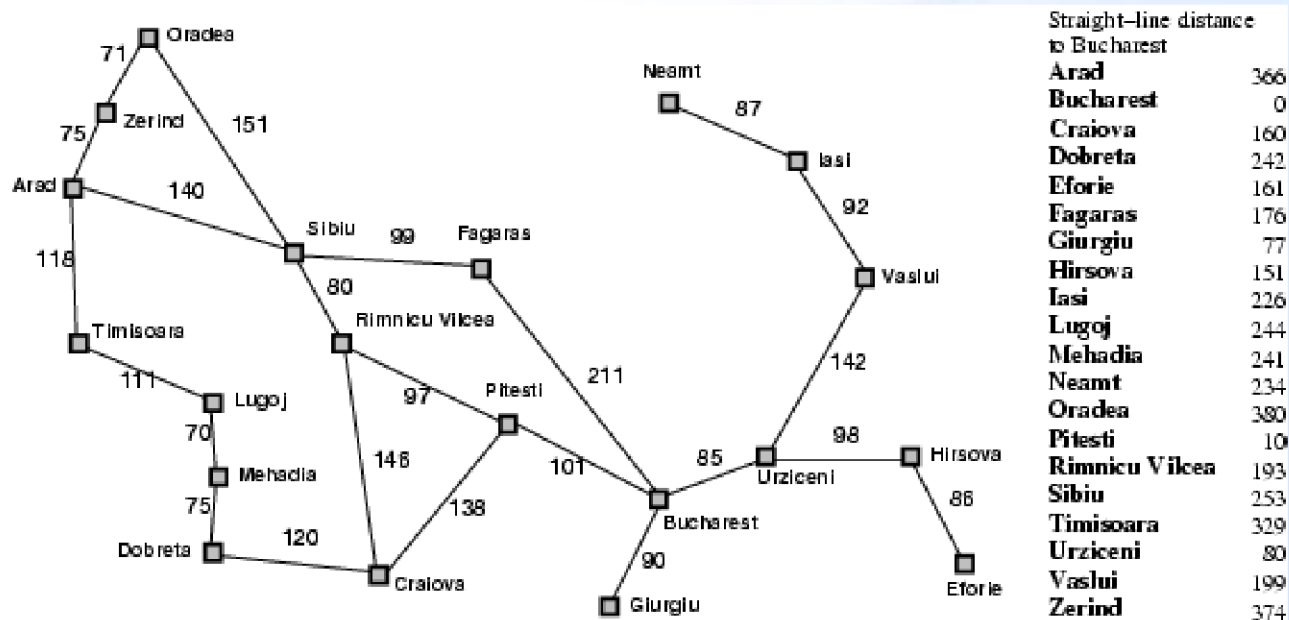
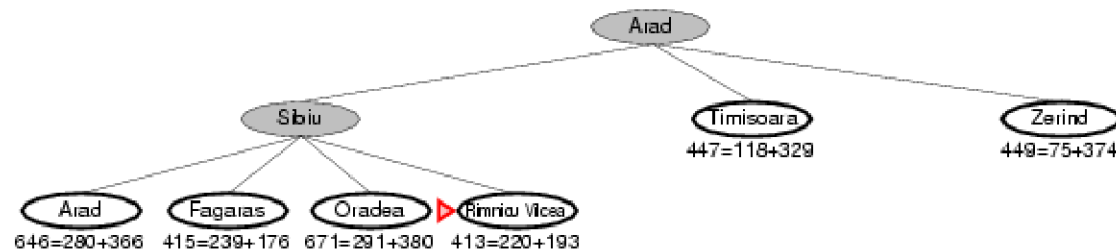
Arad
366=0+366



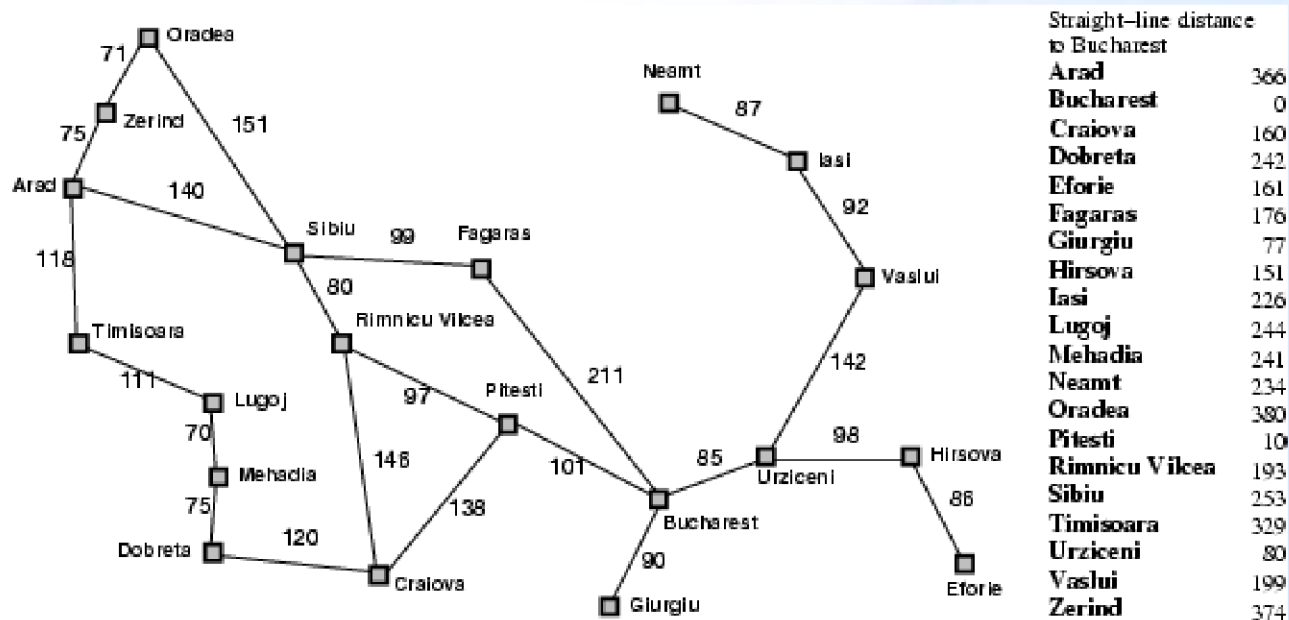
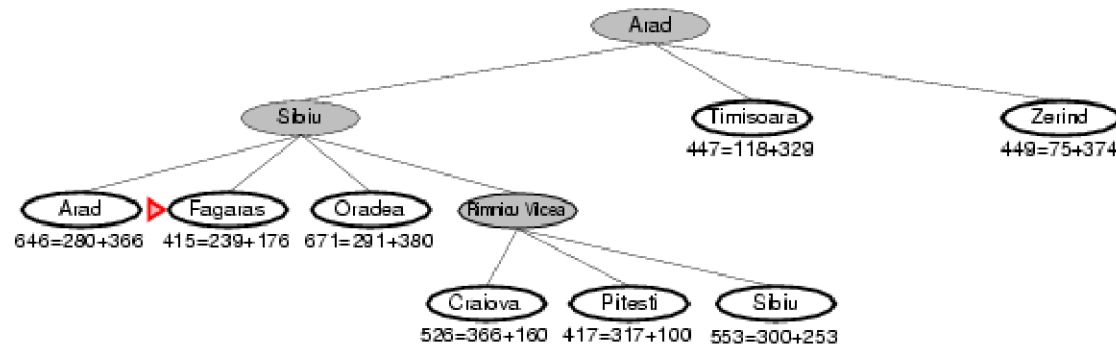
A* search example



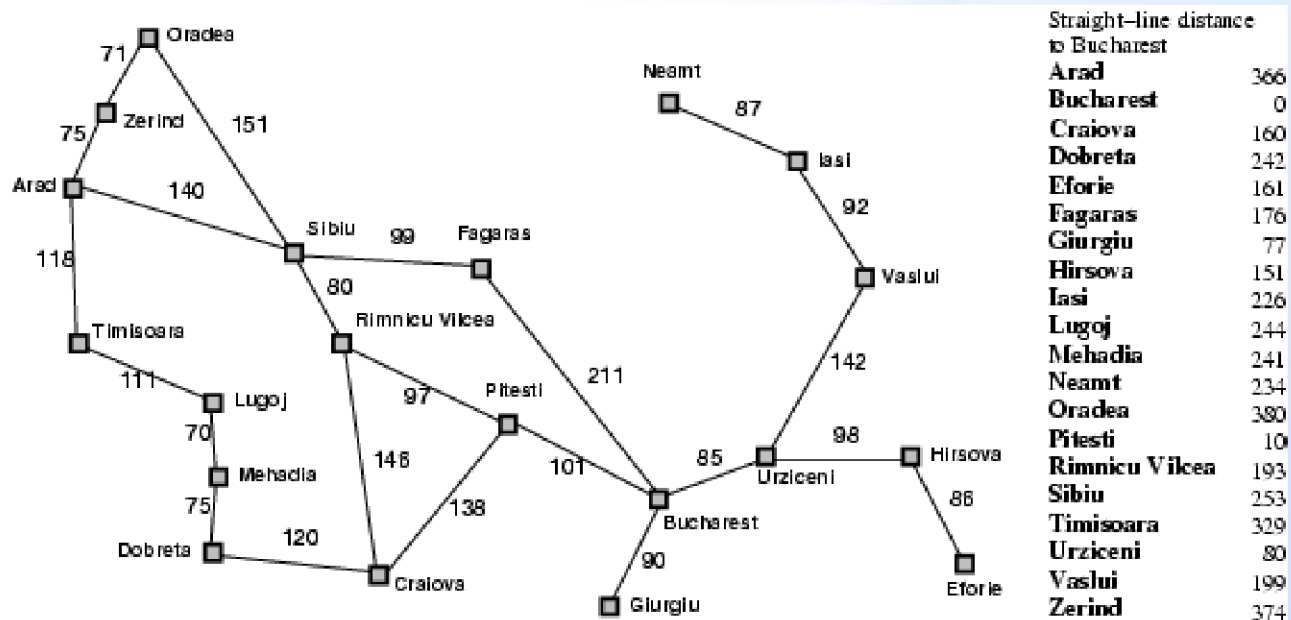
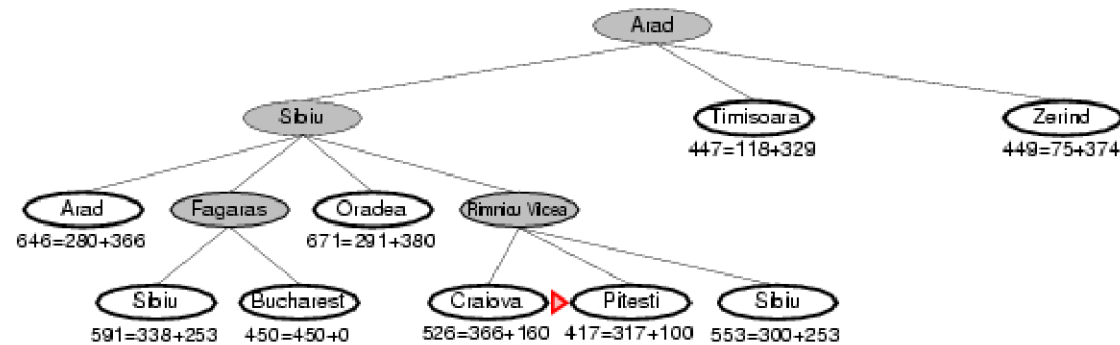
A* search example



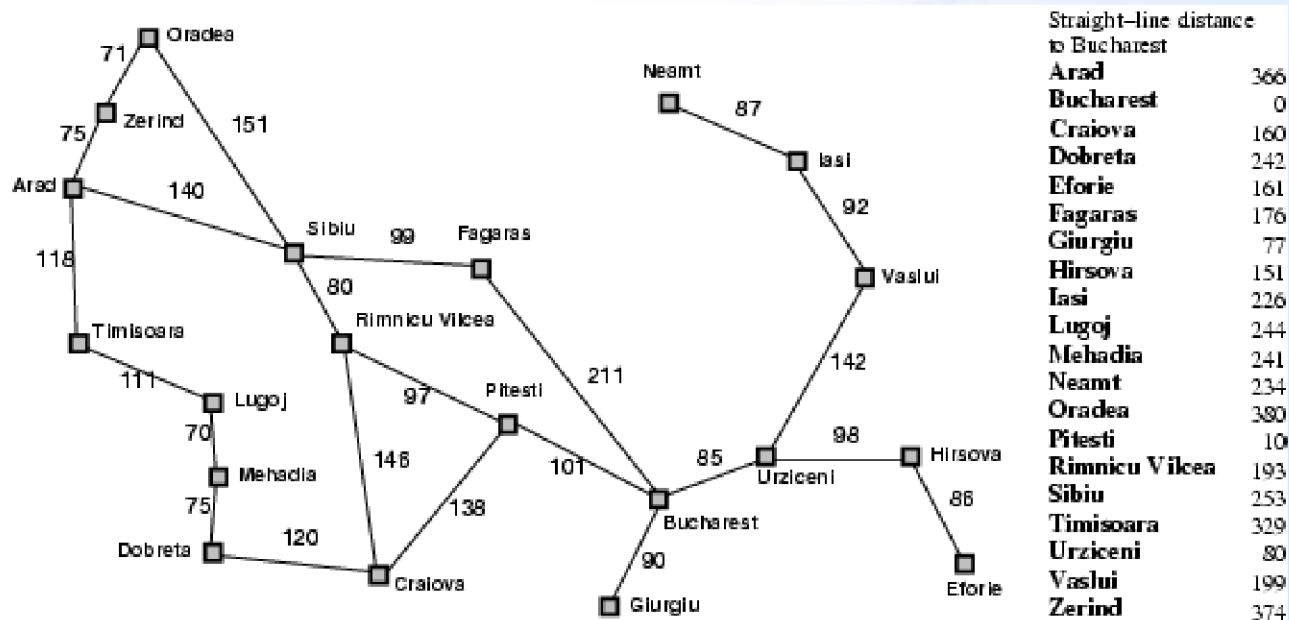
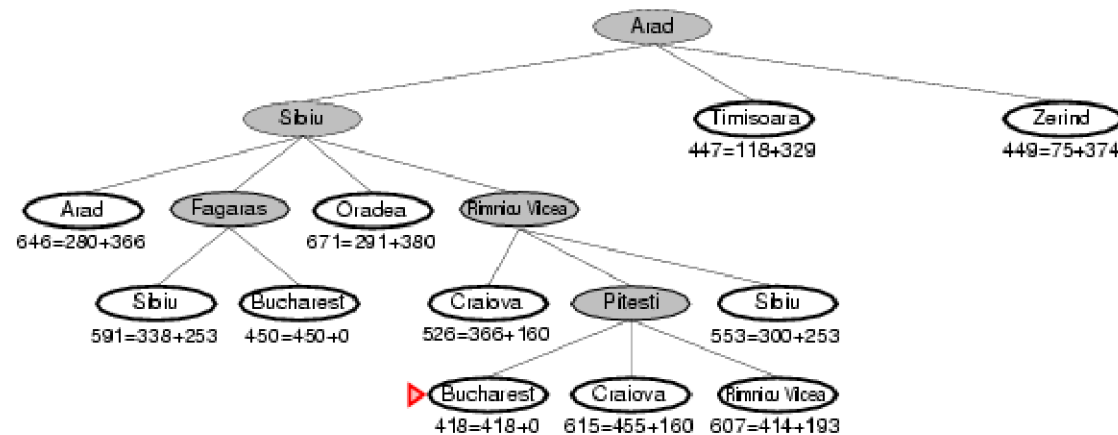
A* search example



A* search example



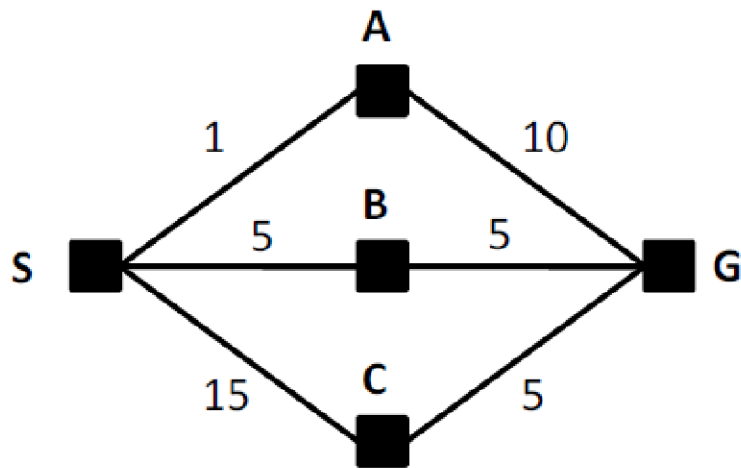
A* search example

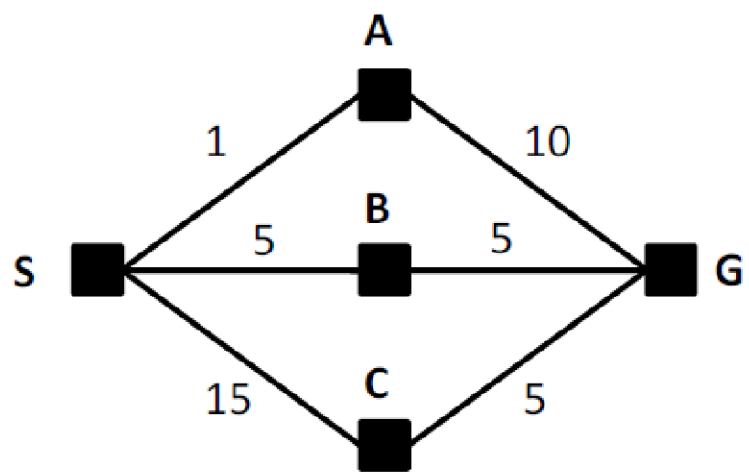


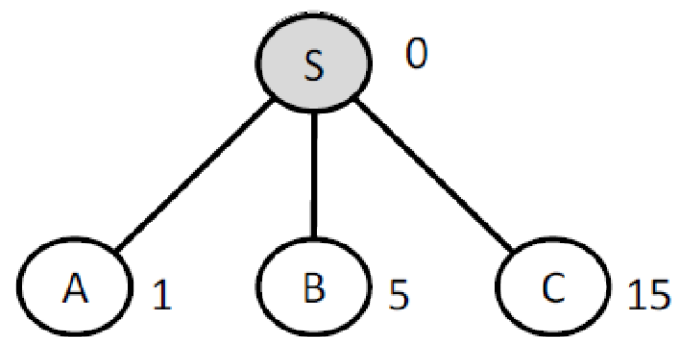
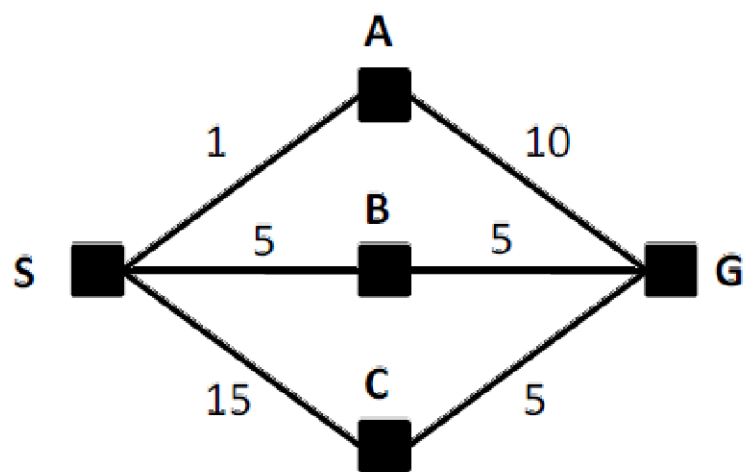
Uniform Cost Search

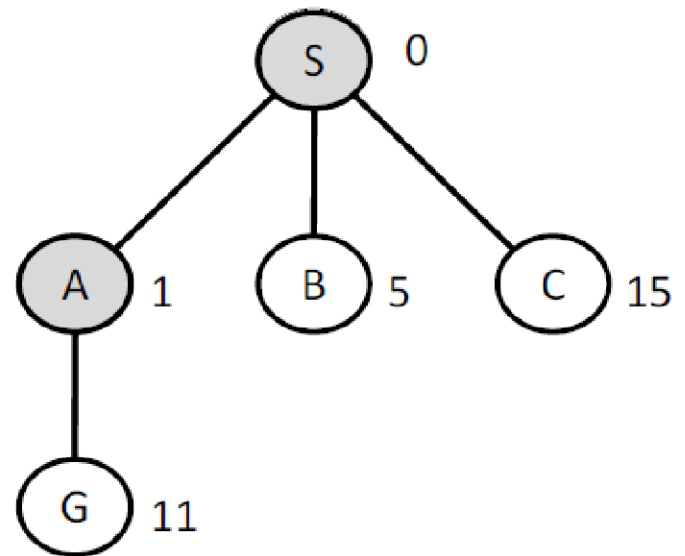
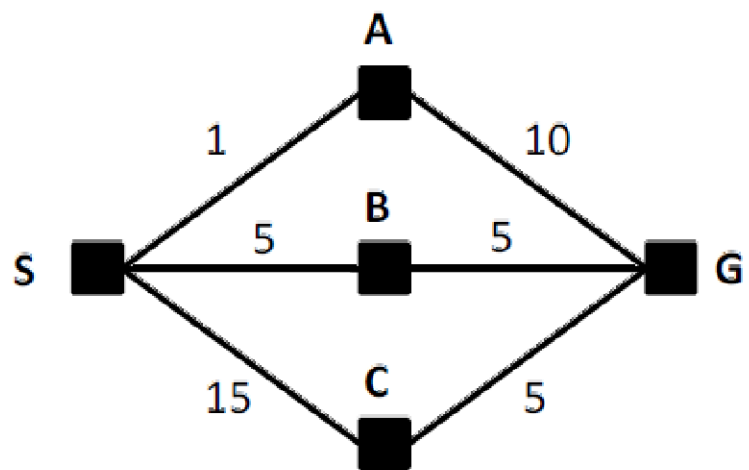
- Prinsip : Lakukan *node expansion* terhadap node yang *path cost-nya* **paling kecil**
- Jika semua step costnya sama, uniform cost sama dengan BFS

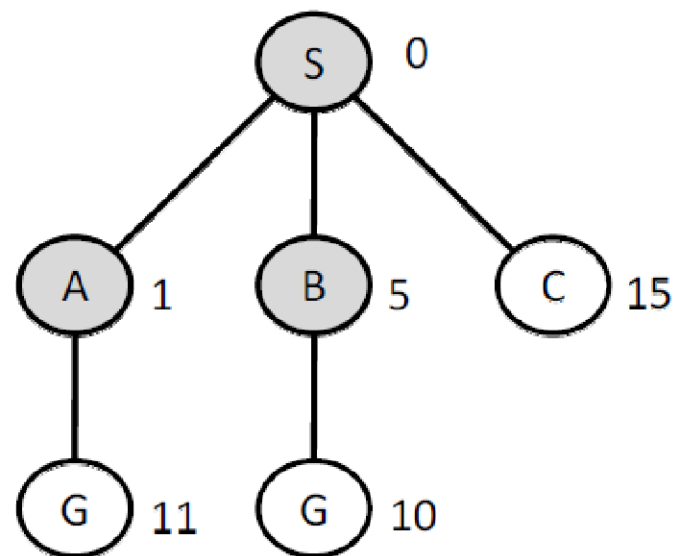
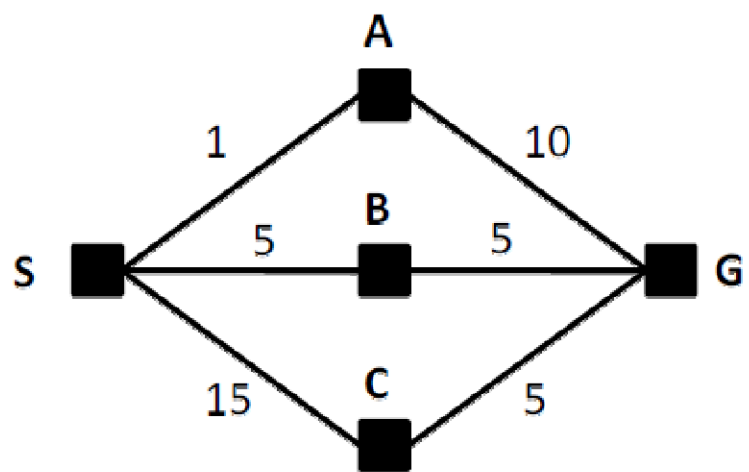
Contoh UCS











Arad ke Bucharest menggunakan UCS

