Introduction to AI

Lecture 6: Informed search strategies

Mona Taghavi

LaSalle College
Montréal

Informed vs uninformed search

- An informed search is a searching technique **that has additional information about the distance from the current state** to the goal.

- An informed search provides the AI with guidance on how and where to look for the solution to the problem. On the other hand, an uninformed search only provides the AI with the problem's specifications and no additional information about the potential solution.

Informed search

Informed search:

Use heuristics to guide the search

- A*
- Heuristics
- Hill-climbing
- Simulated annealing
- Genetic Algorithm
- Beam
- Etc.

Best-first search

- Idea:

  use an evaluation function for each node; estimate of ***"desirability"***

$\Rightarrow$ expand most desirable unexpanded node.

- Implementation:

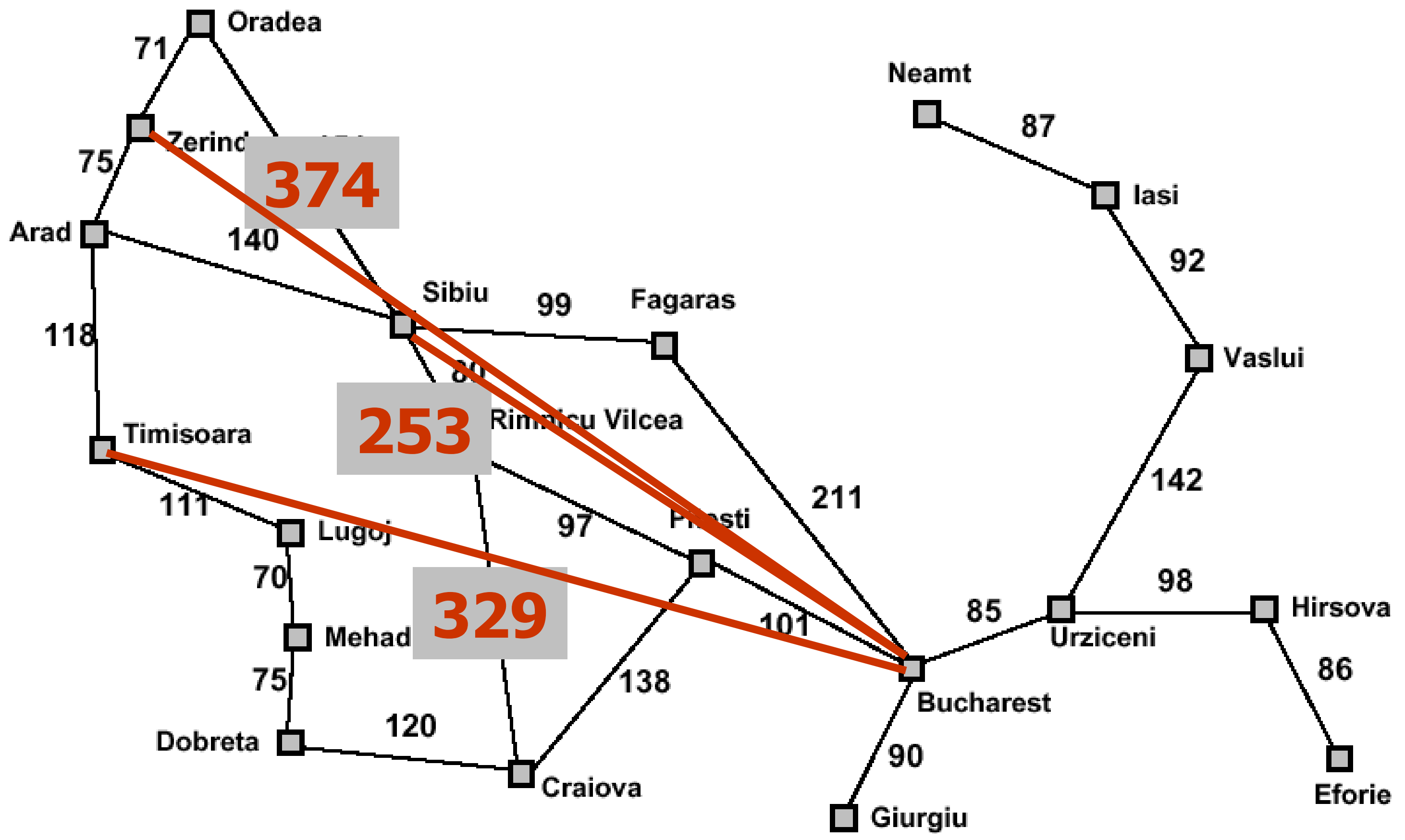
  QueueingFn = insert successors in decreasing order of desirability

- Special cases:

  greedy search

  A* search

Romania with step costs in km



Straight–line distance to Bucharest

Arad	366
Bucharest	0
Craiova	160
Dobreta	242
Eforie	161
Fagaras	178
Giurgiu	77
Hirsova	151
Iasi	226
Lugoj	244
Mehadia	241
Neamt	234
Oradea	380
Pitesti	98
Rimnicu Vilcea	193
Sibiu	253
Timisoara	329
Urziceni	80
Vaslui	199
Zerind	374

Greedy search

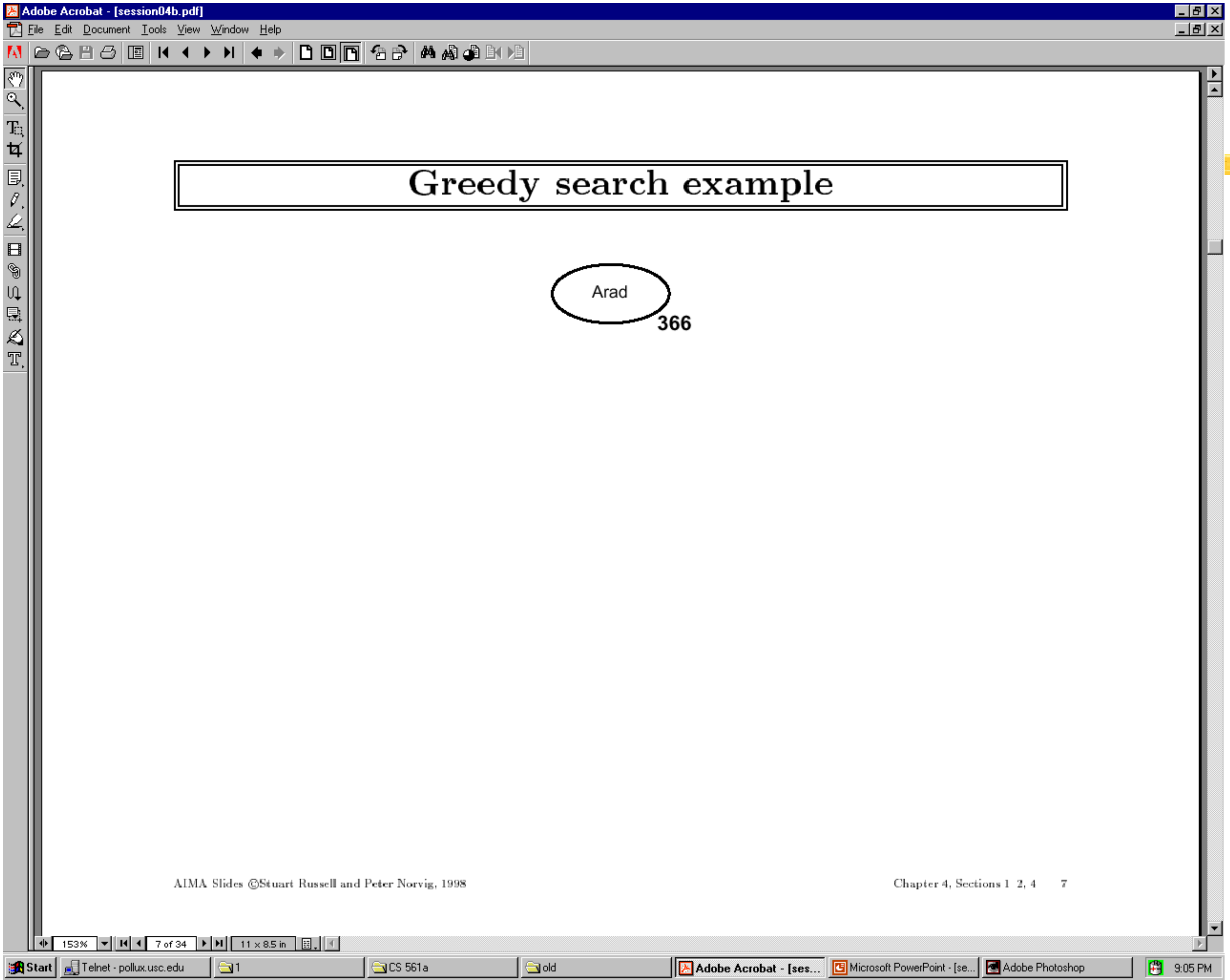
- Estimation function:

  $h(n)$ = estimate of distance from n to goal(heuristic)

- For example:

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

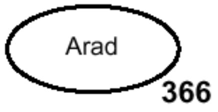
- Greedy search expands first the node that appears to be closest to the goal, according to $h(n)$.



Adobe Acrobat - [session04b.pdf]

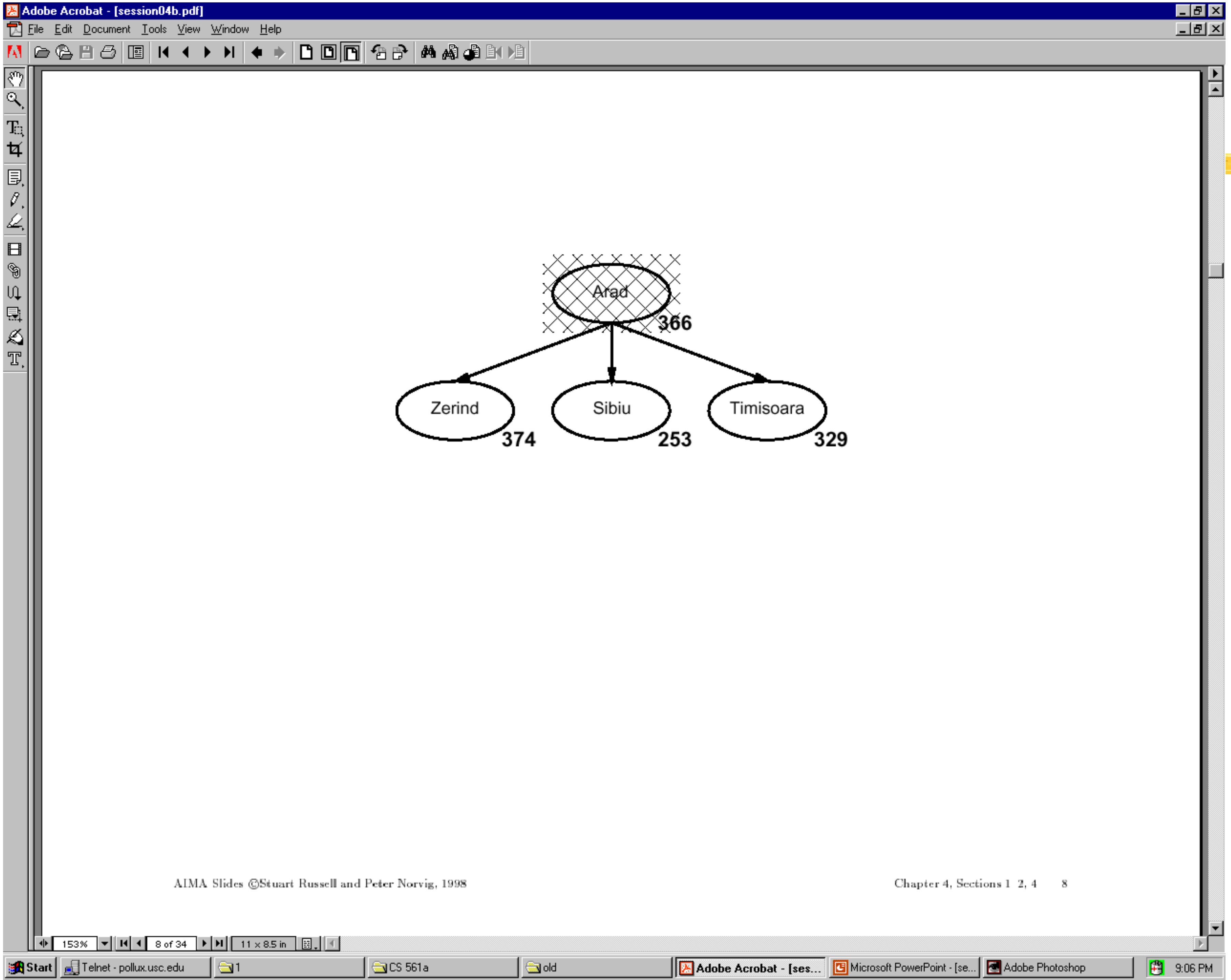
File Edit Document Tools View Window Help

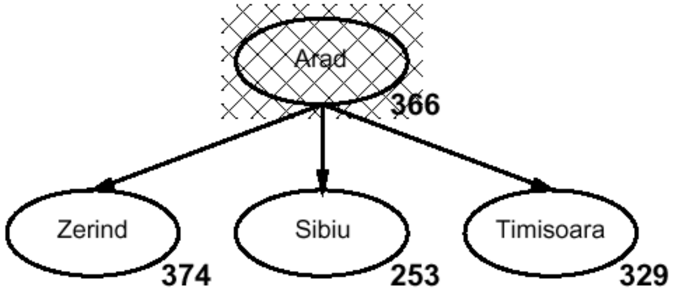
Greedy search example



153% 7 of 34 11 x 8.5 in

Start Telnet - pollux.usc.edu 1 CS 561a old Adobe Acrobat - [ses... Microsoft PowerPoint - [se... Adobe Photoshop 9:05 PM

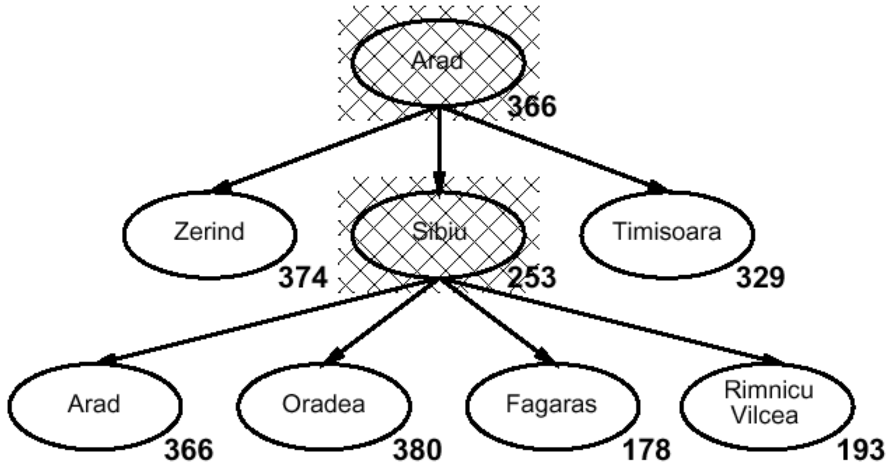

Adobe Acrobat - [session04b.pdf]
File Edit Document Tools View Window Help


Arad
366
Zerind
374
Sibiu
253
Timisoara
329

153%
8 of 34
11 x 8.5 in
Start
Telnet - pollux.usc.edu
1
CS 561a
old
Adobe Acrobat - [ses...
Microsoft PowerPoint - [se...
Adobe Photoshop
9:06 PM

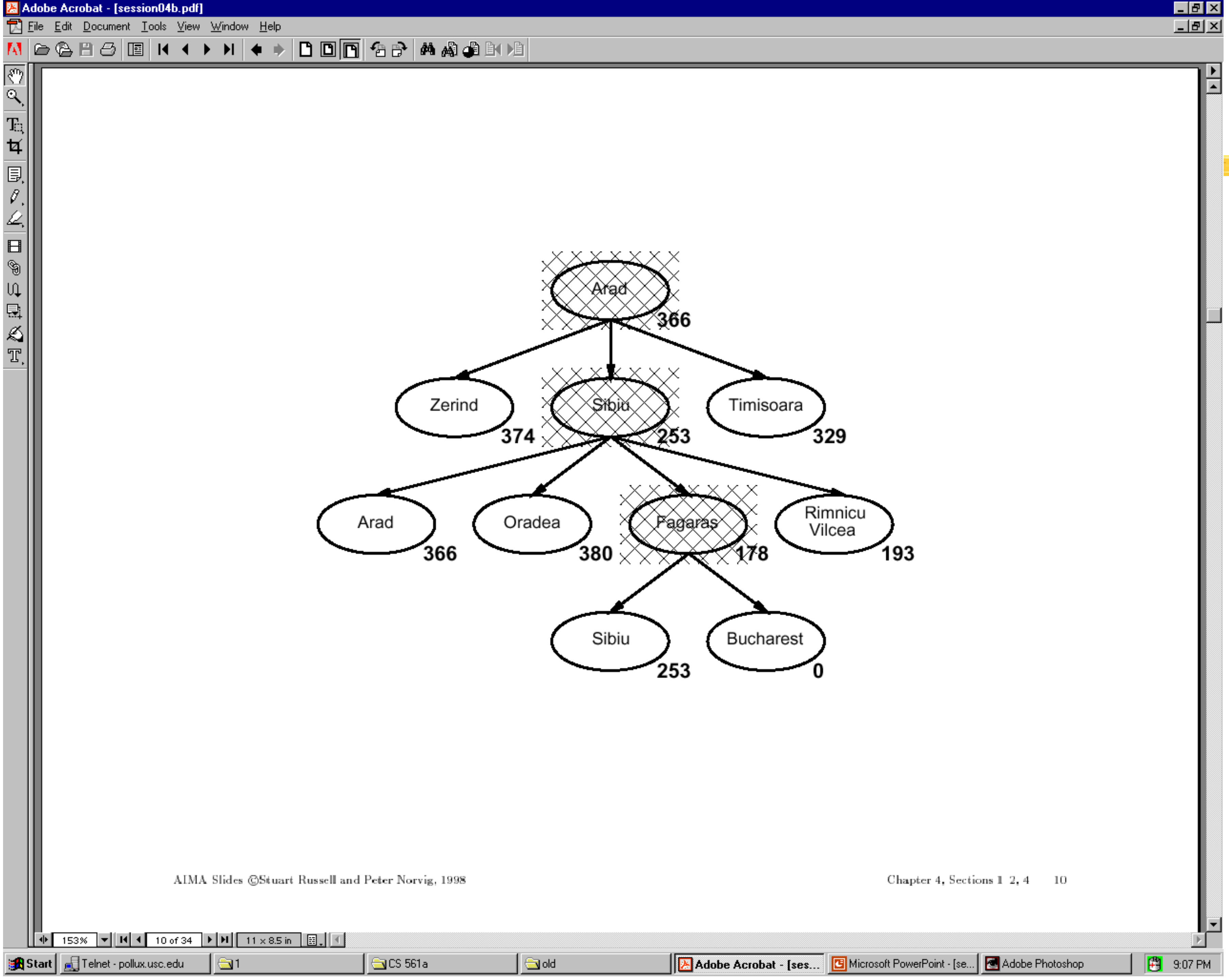
Adobe Acrobat - [session04b.pdf]

File Edit Document Tools View Window Help



153% 9 of 34 11 x 8.5 in

Start | Telnet - pollux.usc.edu | 1 | CS 561a | old | Adobe Acrobat - [ses... | Microsoft PowerPoint - [se... | Adobe Photoshop | 9:06 PM


Adobe Acrobat - [session04b.pdf]
File Edit Document Tools View Window Help
Arad
366
Zerind
374
Sibiu
253
Timisoara
329
Arad
366
Oradea
380
Fagaras
178
Rimnicu Vilcea
193
Sibiu
253
Bucharest
0
153%
10 of 34
11 x 8.5 in
Start
Telnet - pollux.usc.edu
1
CS 561a
old
Adobe Acrobat - [ses...
Microsoft PowerPoint - [se...
Adobe Photoshop
9:07 PM

Measuring Distance

Euclidean Distance

Manhattan Distance

Chebyshev Distance

A* search

- Idea: avoid expanding paths that are already expensive

  evaluation function: $f(n) = g(n) + h(n)$ with:

  $g(n)$ – cost so far to reach n

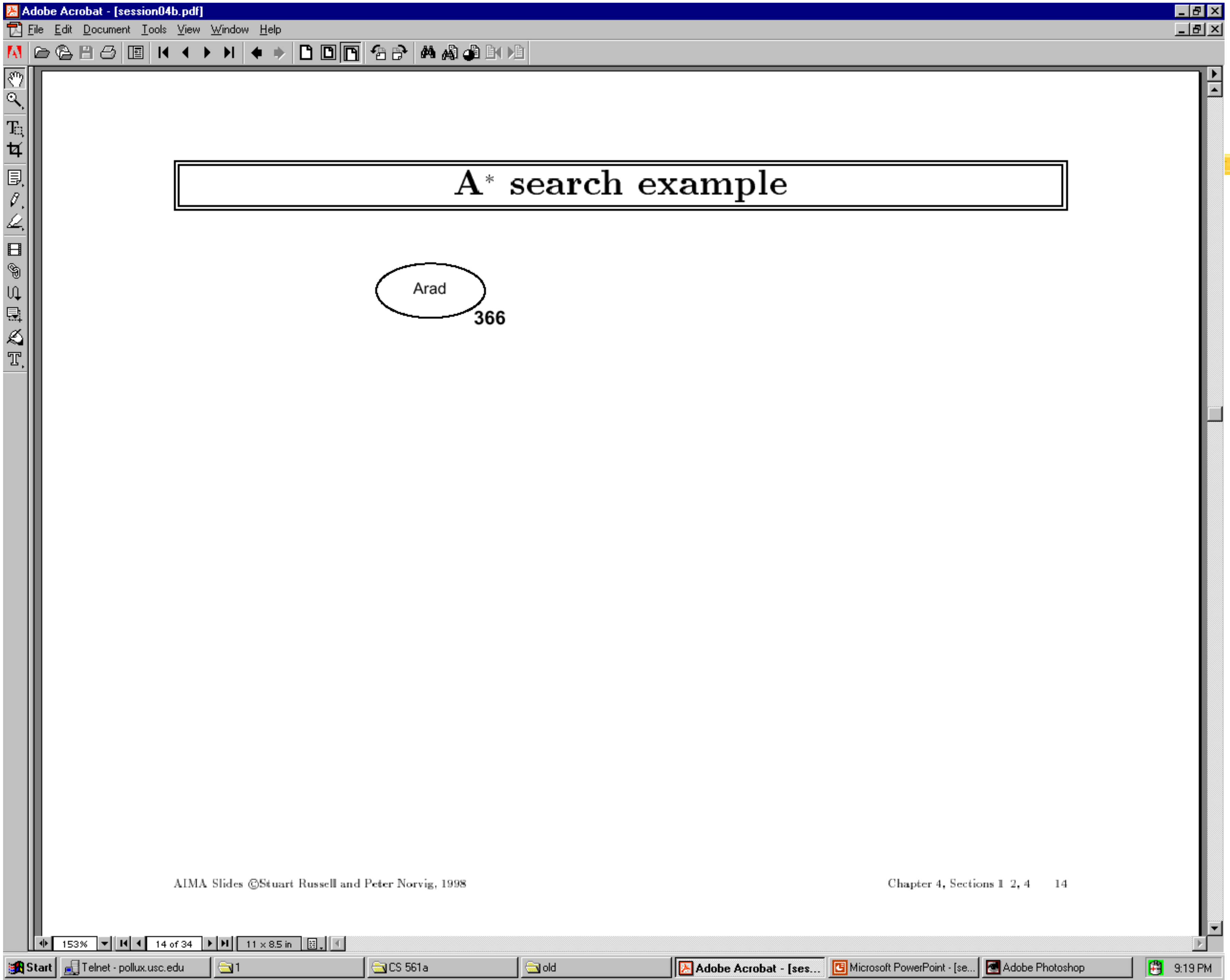
  $h(n)$ – estimated cost to goal from n

  $f(n)$ – estimated total cost of path through n to goal

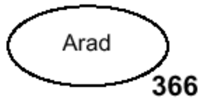
- A* search uses an admissible heuristic, that is,

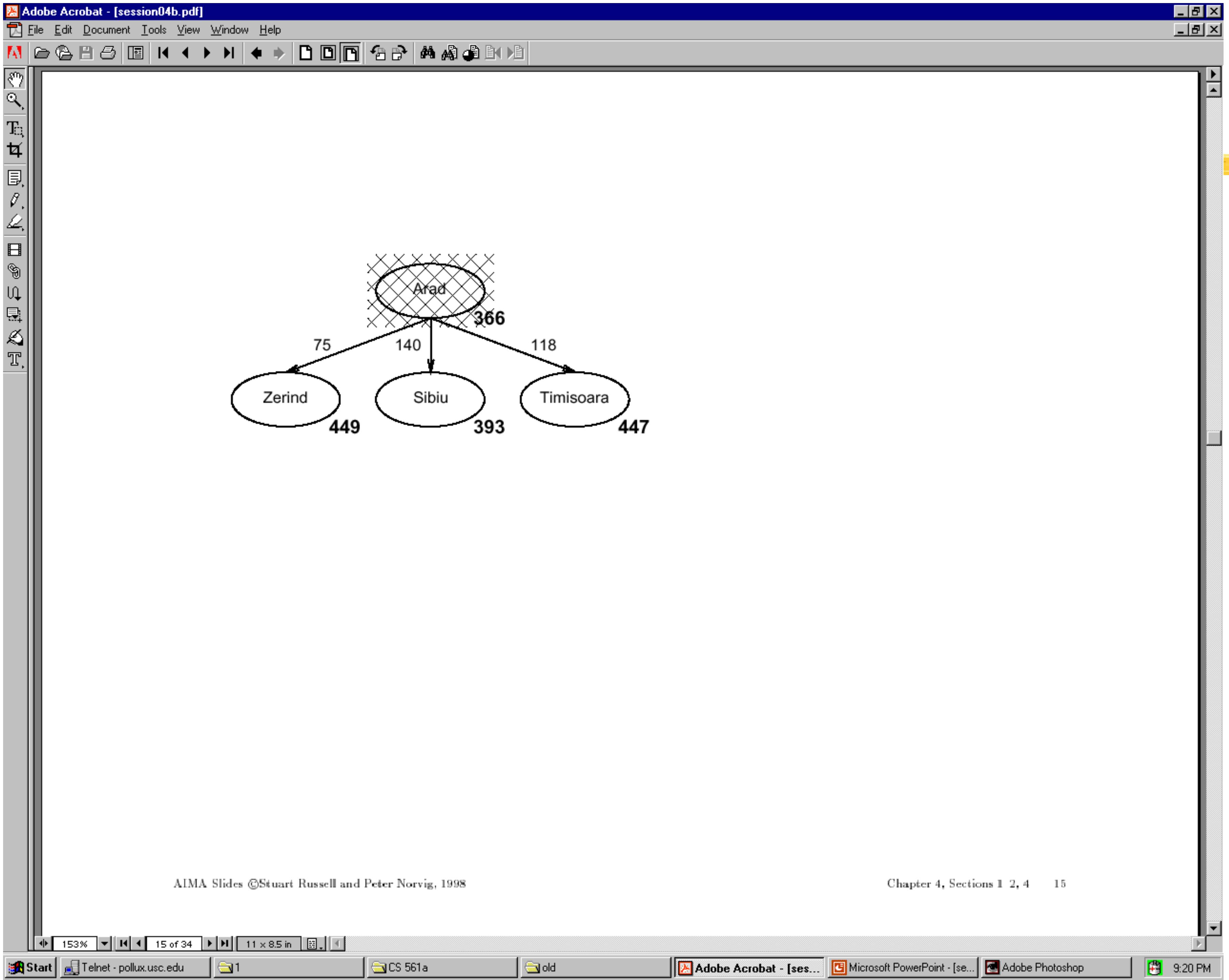
  $h(n) \leq h^*(n)$ where $h^*(n)$ is the true cost from n.

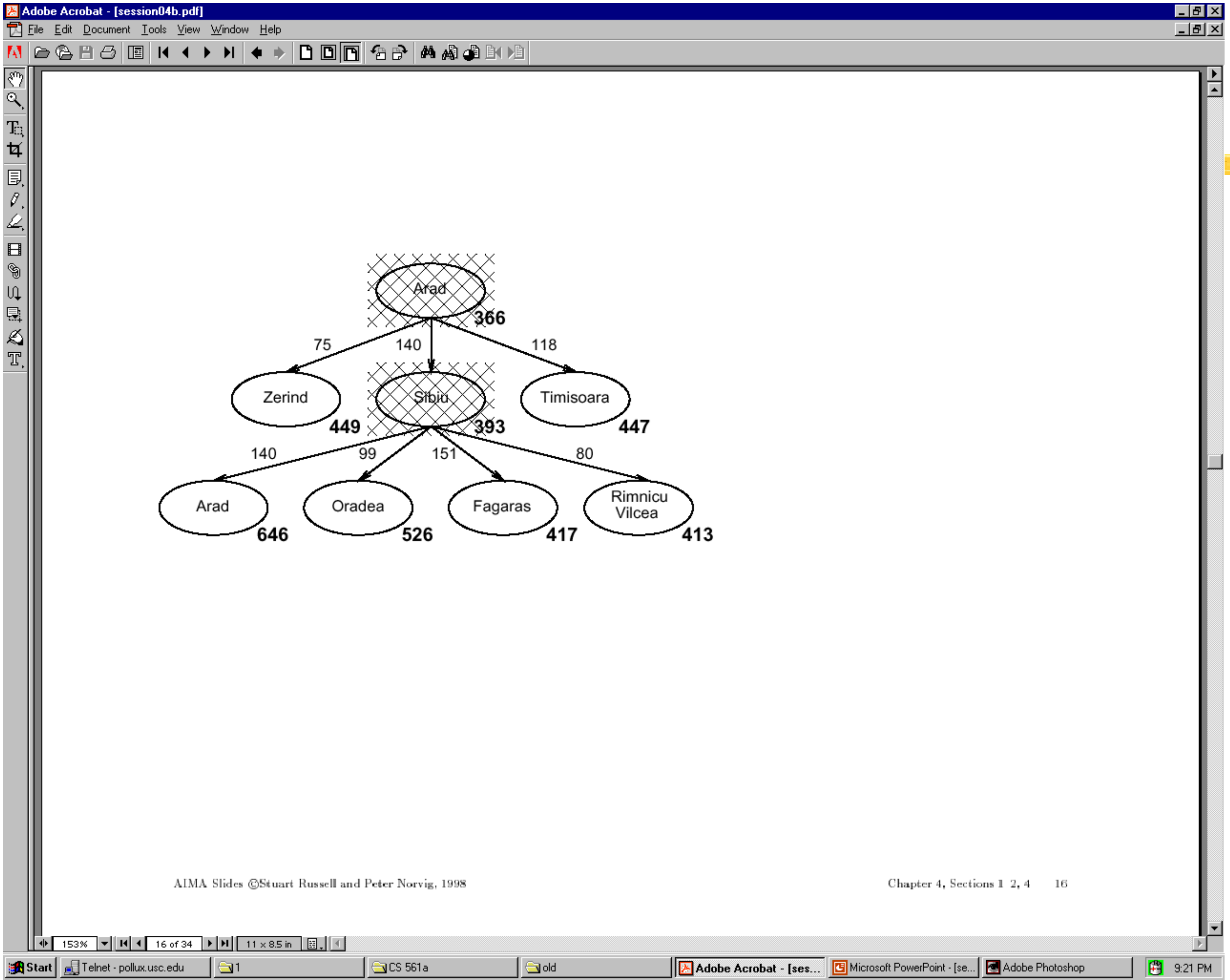
  For example: $h_{SLD}(n)$ never overestimates actual road distance.



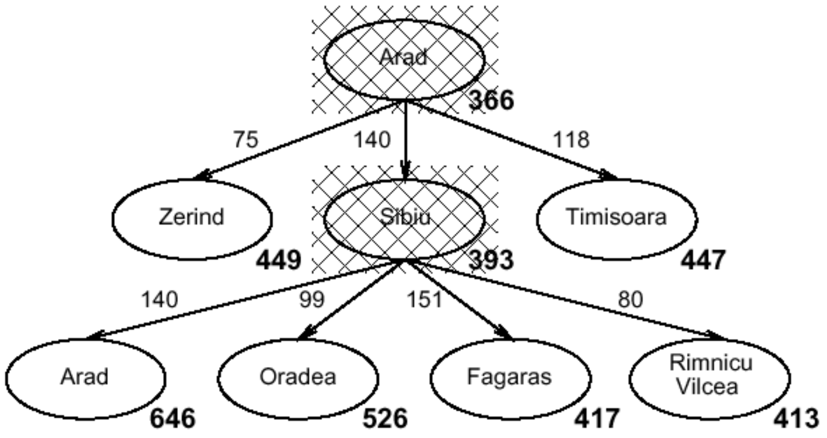
A* search example

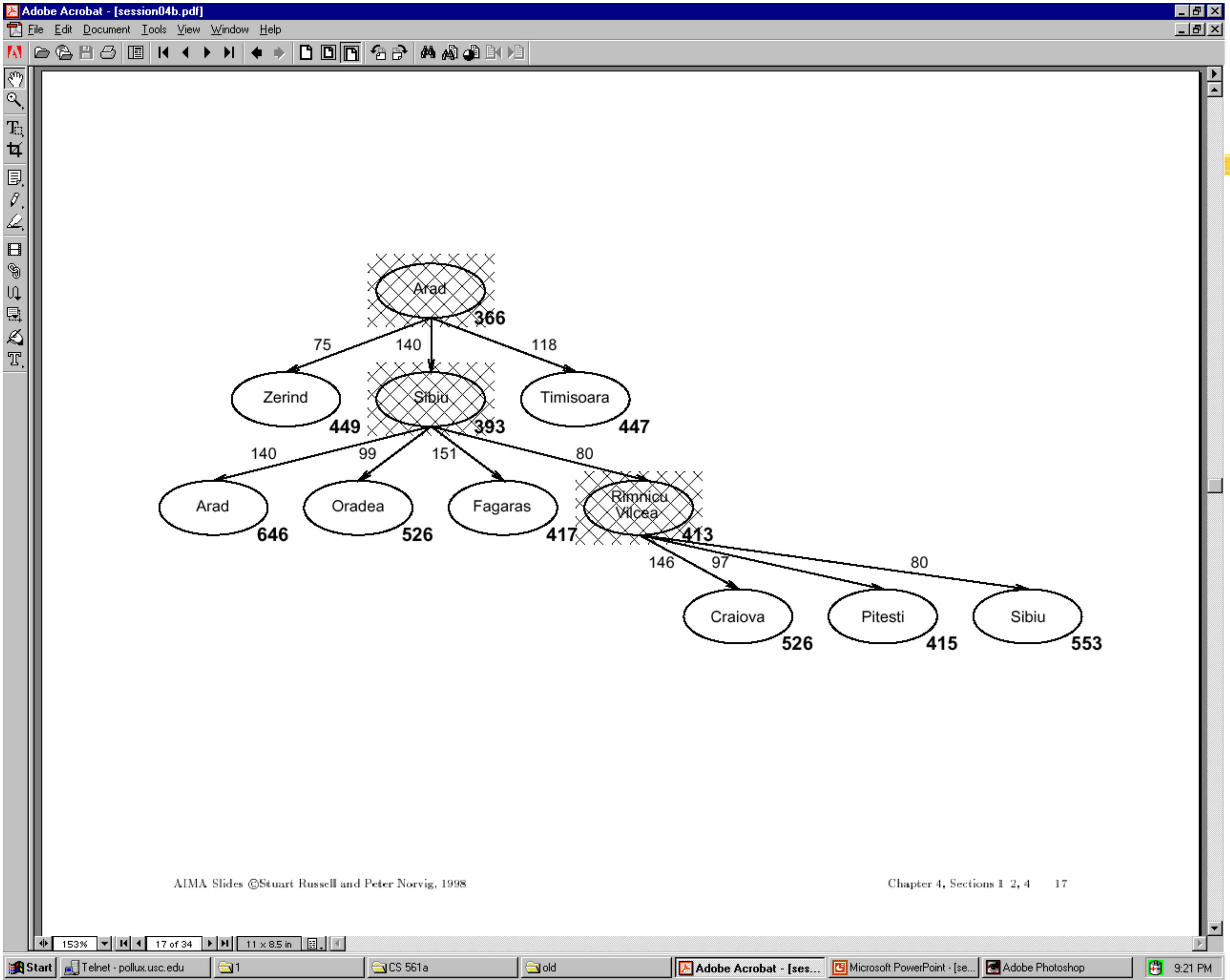



Adobe Acrobat - [session04b.pdf]
File Edit Document Tools View Window Help
Arad
366
75
140
118
Zerind
449
Sibiu
393
Timisoara
447
153%
15 of 34
11 x 8.5 in
Start
Telnet - pollux.usc.edu
1
CS 561a
old
Adobe Acrobat - [ses...
Microsoft PowerPoint - [se...
Adobe Photoshop
9:20 PM

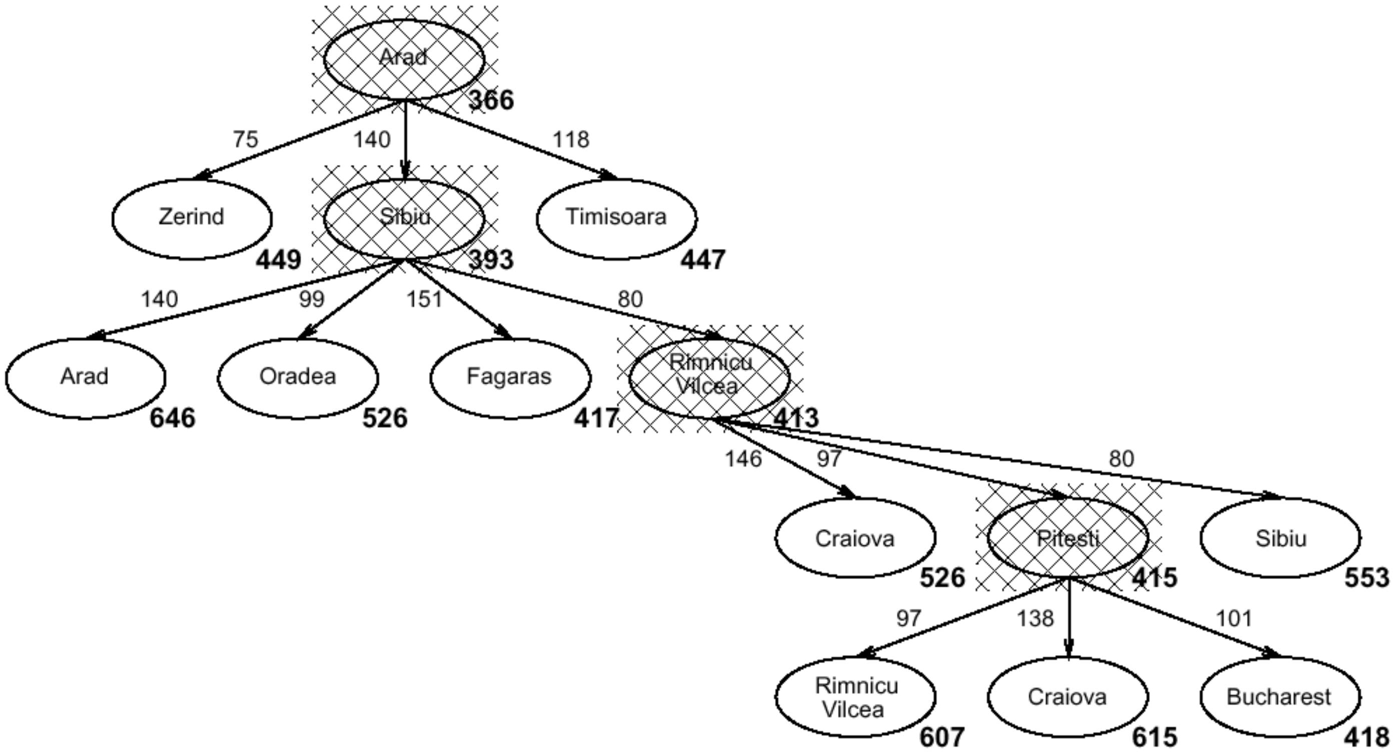


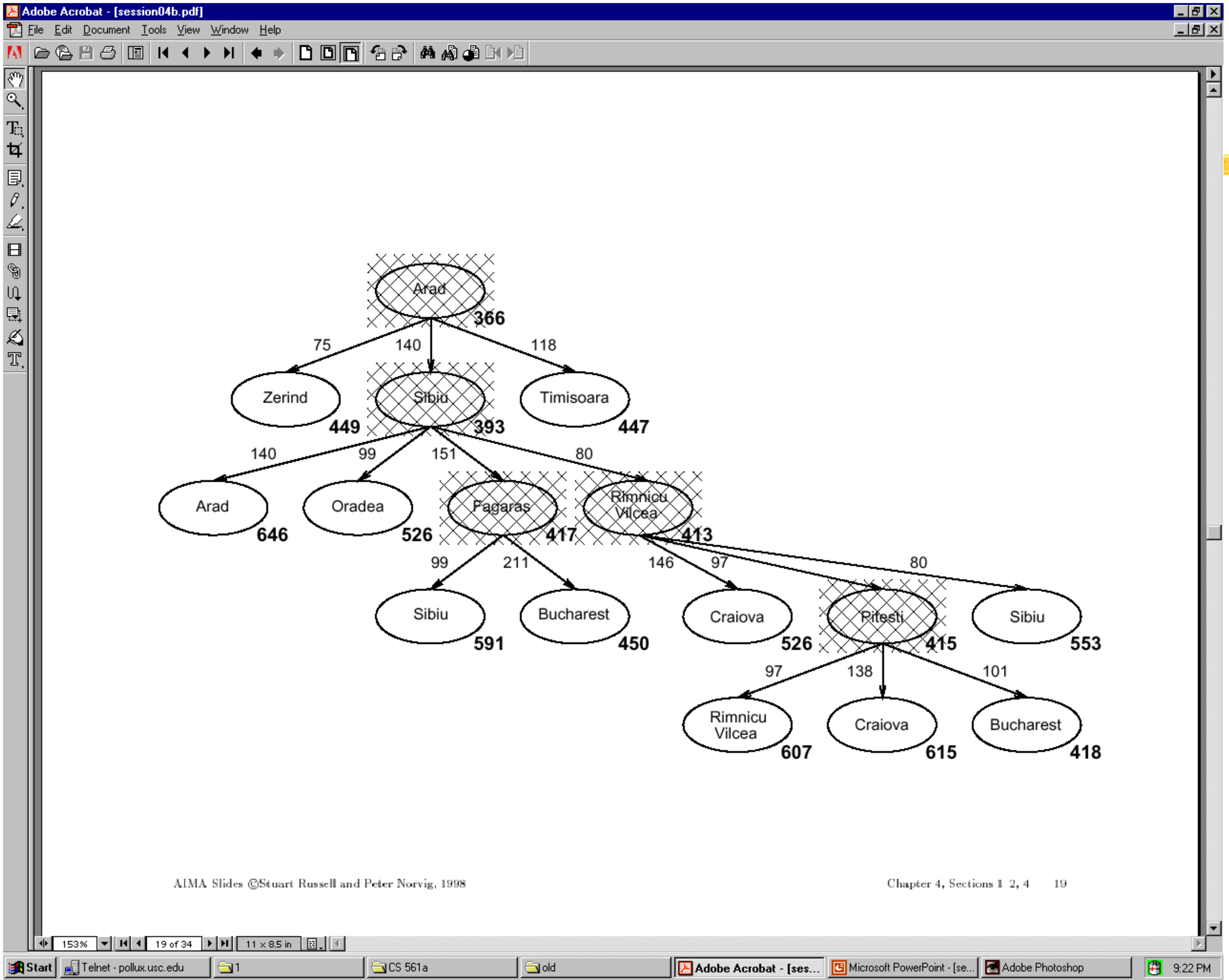
File Edit Document Tools View Window Help




Adobe Acrobat - [session04b.pdf]
File Edit Document Tools View Window Help
Arad
366
75
140
118
Zerind
449
Sibiu
393
Timisoara
447
140
99
151
80
Arad
646
Oradea
526
Fagaras
417
Rimnicu Vilcea
413
146
97
80
Craiova
526
Pitesti
415
Sibiu
553
153%
17 of 34
11 x 8.5 in
Start
Telnet - pollux.usc.edu
1
CS 561a
old
Adobe Acrobat - [ses...
Microsoft PowerPoint - [se...
Adobe Photoshop
9:21 PM

Adobe Acrobat - [session04b.pdf]




Adobe Acrobat - [session04b.pdf]

Weighted A* Search

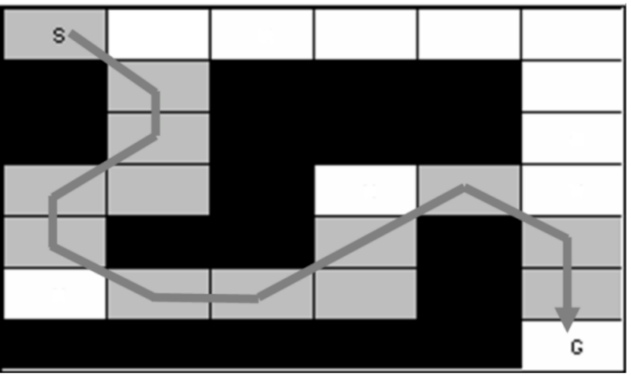
- Expands states in the order of $f(n) = g(n) + \varepsilon h(n)$ values
  - $\varepsilon > 1$ = bias towards states that are closer to goal

- Trades off optimality for speed
- In games we often multiply the heuristic by some constant factor to make A* run faster.

Effect of the Heuristic Function

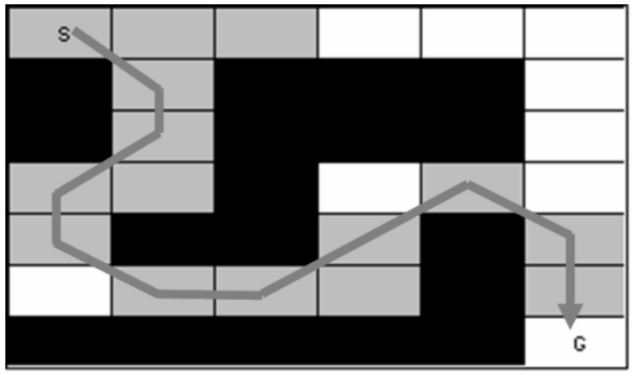
- Constructing anytime search based on weighted A*:
  - find the best path possible given some amount of time for planning
  - do it by running a series of weighted A* searches with decreasing ε

$\varepsilon = 2.5$



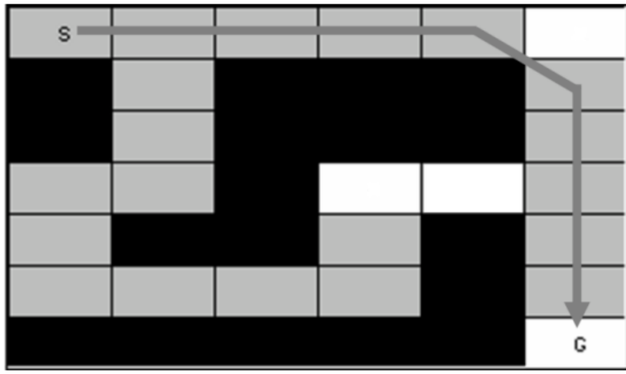
13 expansions
solution=11 moves

$\varepsilon = 1.5$



15 expansions
solution=11 moves

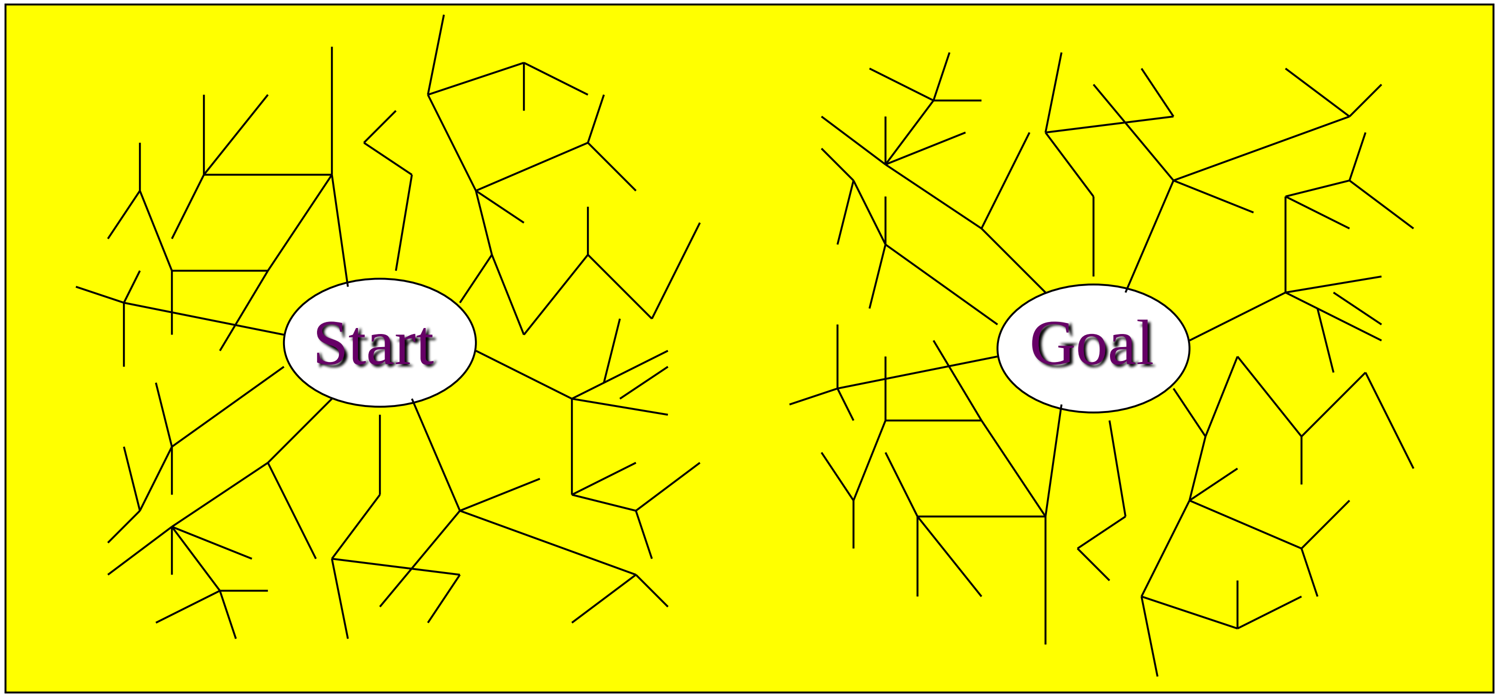
$\varepsilon = 1.0$



20 expansions
solution=10 moves

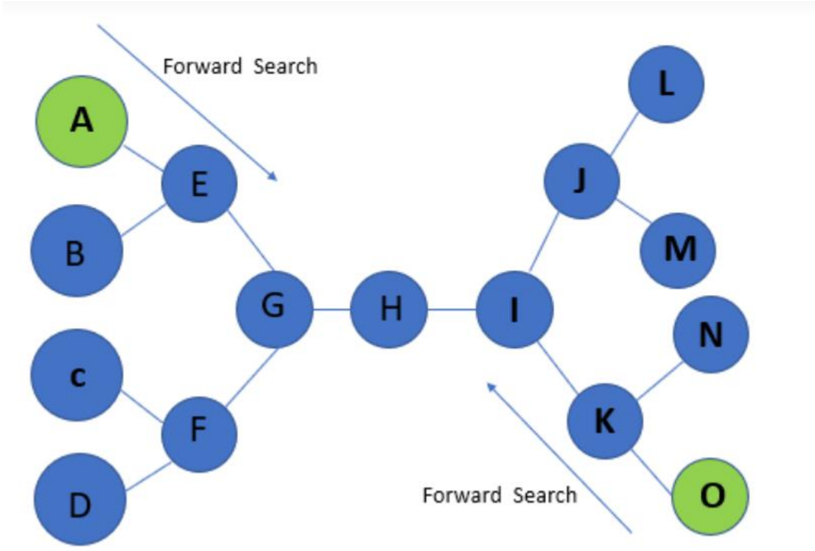
Bi-directional search (uninformed/informed search)

- Compute the tree both from the start-node and from a goal node, until these meet.

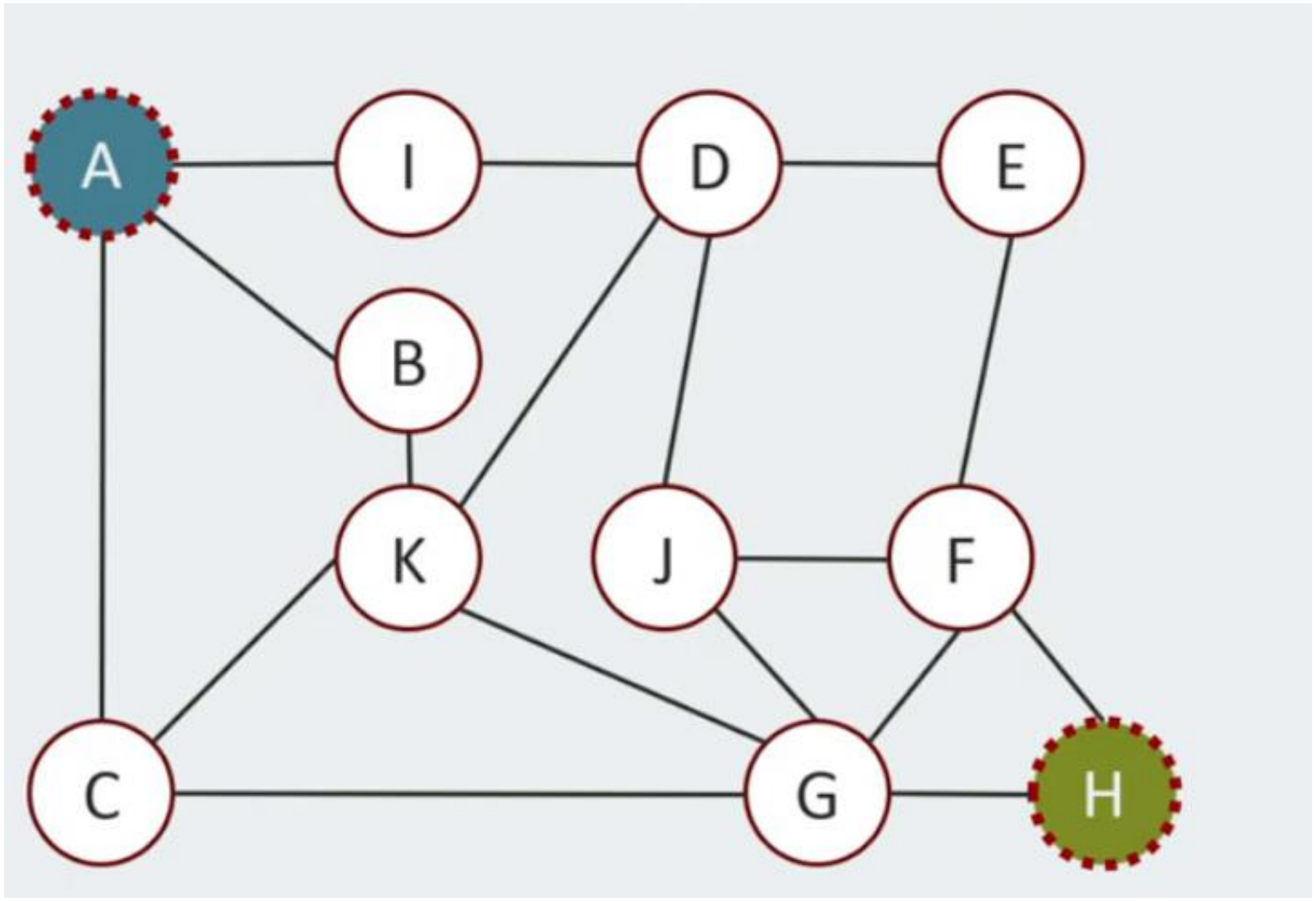


Bi-directional search

- Instead of searching from the start to the finish, you can start two searches in parallel—one from start to finish, and one from finish to start. When they meet, you should have a good path.
- The main idea behind bidirectional searches is to reduce the time taken for search drastically.
- It’s a good idea that will help in some situations. The idea behind bidirectional searches is that searching results in a “tree” that fans out over the map. A big tree is much worse than two small trees, so it’s better to have two small search trees.



Use BFS in Bidirectional search to move from A to H


A
I
D
E
B
K
J
F
C
G
H