Title: Solving Problems by Searching

AIMA: Chapter 3 (Section 3.6)

Introduction to Artificial Intelligence

CSCE 476-876, Spring 2001

URL: `www.cse.unl.edu/˜choueiry/S01-476-876`

Berthe Y. Choueiry (Shu-we-ri)

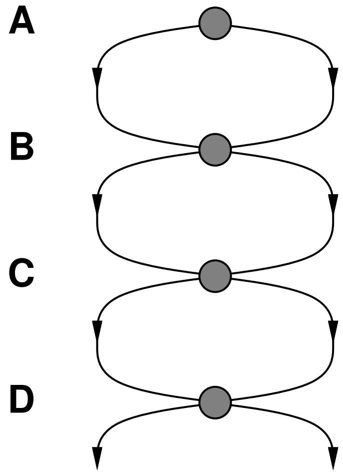
`choueiry@cse.unl.edu, (402)472-5444`

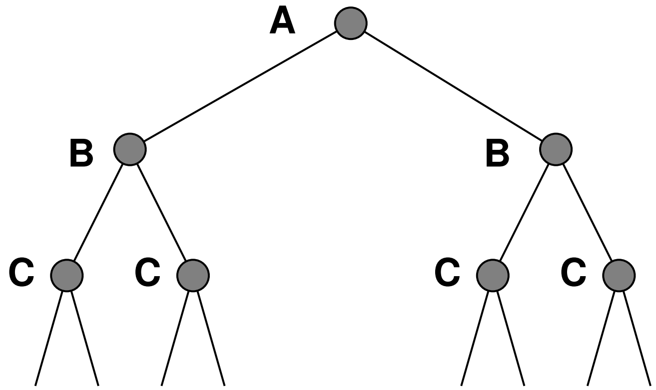
Loops: Avoid repeated states (I)

Avoid expanding states that have already been visited

Valid for both infinite and finite trees

Example: $\begin{cases} m \text{ maximum depth} \\ m+1 \text{ states} \\ 2^m \text{ possible branches (paths)} \end{cases}$



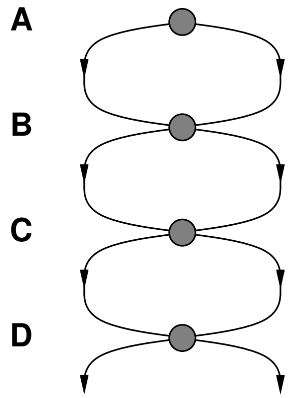


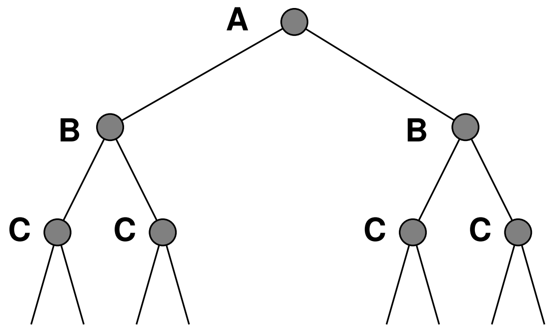
Loops: (II)

3 Strategies:

1. Do not return to the state you come from
2. Do not return to any of ancestor state
3. Do not return to any previously visited state

   ...... required space?





$\longrightarrow$ Implementation: hash table.

Trade-off cost of storing+checking vs. cost of searching

Summary

Path: sequence of actions leading from one state to another

Partial solution: a path from an initial state to another state

Search: develop a sets of partial solutions

- Search tree & its components (node, root, leaves, fringe)
- Data structure for a search node
- Search space vs. state space
- Node expansion, queue order
- Search types: uniformed vs. heuristic
- 6 uniformed search strategies
- 4 criteria for evaluating & comparing search strategies