Admin:
- Quiz: Wed, night 7:30-9:30 in 1-190 (conflict: Th 8am-11pm - see TA's) 10/14/2008
  bring one sheet of notes
- No recitations tomorrow
Readings: CLRS: 22.1-22.3, B.4

Outline: Search (1 of 3)
- graphs review & apps
- graph representations
- intro to BFS (breadth-first search) & DFS (depth-first search)

Graph search: to explore a graph.
  to find a path from a start vertex s to a desired target vertex

Recall: graph $G = (V, E)$

where $V =$ (finite) set of vertices
$E =$ set of edges (vertex pairs) $\subseteq V \times V$
if $G$ is directed graph, then edge is ordered pair $(u, v)$
(i.e. from $u$ to $v$) $u \rightarrow v$ "can get from $u$ to $v"$

if $G$ is undirected graph, then edge is unordered pair $\{u, v\}$
representing connection between $u \& v$ both ways
$u \leftrightarrow v$ "can go both ways"
Examples

Undirected graph

\[ V = \{a, b, c, d\} \]
\[ E = \{\{a, b\}, \{a, c\}, \{b, c\}, \{b, d\}, \{c, d\}\} \]

Directed graph

\[ V = \{a, b, c\} \]
\[ E = \{(a, c), (b, c), (c, b), (b, a)\} \]

Note: self-loops & multiple edges typically excluded
often write \((u, v)\) for edge even if \(G\) is undirected
(understand that it means unordered pair, though)

Graph has two parameters describing its size:
\[ |V| = \# \text{ of vertices} \]
\[ |E| = \# \text{ of edges} \quad (0 \leq |E| < |V|^2) \]
Running times may be described as function of both.
Applications: many!!!
- web crawling (how Google finds pages)
- internet routing
- solving puzzles & games

Rivest
L12.3
10/14/08

Pocket Cube (2x2x2 Rubik's cube)
- defines configuration graph:
  - vertex for each possible state
  - edge for each basic move (e.g. 90° turn) (180° ?? too) from one state to another
- puzzle: given some initial state s,
  - find path to solved state
- $|V| = 8! \cdot 3^8 = 264,539,520$
  - rotate cubelet
  - 8 cubelets in arbitrary positions
- This is big! Can we simplify?
- Factor out 24-fold symmetry: fix position & orientation of one cubelet
  - $|V| = 7! \cdot 3^7 = 11,022,480$
- In fact, graph has 3 "connected components"
  - only 1/3 of states are reachable
    - (without disassembling cube)
    - Only need to search one component
  - $|V| = 7! \cdot 3^6 = 3,674,160$
Breadth-first search of configuration graph

\[ \text{distance} \quad 90^\circ \text{ turns} \quad 90^\circ \& 180^\circ \text{ turns} \]

\[ \begin{array}{ccc}
0 & 1 & 1 \\
1 & 6 & 9 \\
2 & 27 & 54 \\
3 & 120 & 321 \\
4 & 534 & 1,847 \\
5 & 2,256 & 9,992 \\
6 & 8,969 & 50,136 \\
7 & 33,058 & 227,536 \\
8 & 114,149 & 870,072 \\
9 & 360,508 & 1,887,748 \\
10 & 930,588 & 623,800 \\
11 & 1,350,852 & 2,644 \leftarrow \text{diameter 11} \\
12 & 782,536 & 0 \\
13 & 90,280 & 0 \\
14 & 276 \leftarrow \text{diameter 14} & 0 \\
\hline
3,674,160 & 3,674,160
\end{array} \]

For 3x3x3 Rubik's cube: \( \approx 1.4 \text{ trillion states} \)

diameter \( \leq 26 \) but unknown.
Graph representations on computer

Operations to support:
- create empty graph
- add vertex or edge
- remove vertex or edge
- list vertices adjacent to given vertex \( u \) \( \text{neighbors} \)
- test if edge \( (u,v) \in E \)

1. Adjacency list representation:
   \[ V = \text{set of vertices (hashable)} \]
   \[ \text{Adj: dictionary} \]
   \[ \text{Adj}[u] = \text{list of vertices adjacent to } u \]
   
   Note: can have multiple graphs on same set of vertices

2. Object-oriented variations:
   \[
   \begin{align*}
   \text{vertex } u & \text{ is an object} \\
   u.\text{neighbors} &= \text{Adj}[u] \text{ neighbors of } u \\
   u.\text{edges} &= \text{list of (outgoing) edges from } u
   \end{align*}
   \]

Above reps are good for sparse graphs, where \( |E| \ll |V|^2 \)

since space requirement is \( O(V+E) \)

\( \leftarrow \) don't bother with 1.1's

inside \( O(\Theta) \)
3. Adjacency matrix representation
Assume $V = \{1, 2, \ldots, |V|\}$
Let $A = (a_{ij}) = |V| \times |V|$ matrix
where $a_{ij} = \begin{cases} 1 & \text{if } (i,j) \in E \\ 0 & \text{otherwise} \end{cases}$
$i =$ row 
$j =$ column

E.g.

$$
\begin{bmatrix}
0 & 0 & 1 \\
1 & 0 & 1 \\
0 & 0 & 0
\end{bmatrix}
$$

Good for dense graphs (where $|E| \approx |V|^2$)
Space required = $\Theta(|V|^2)$
Cool properties: $A^3$ gives length-3 paths
Can test quickly if $(i,j) \in E$

4. Implicit graphs
$\text{Adj}(u)$ is a function returning list of neighbors
No space needed! (except during search, to keep track of places visited...)
Can do this for Pocket Cube
High-level overview of next two lectures:

Breadth-first search:

- Explor “level-by-level”
- Mark vertices once discovered (need space |V|) - relevant to Pocket Cube
- All unmarked vertices adjacent to level k \rightarrow level k+1, mark them.
- Finds shortest paths.

Depth-first search:

- Like exploring a maze
- Backtrack when necessary
- To find unexplored edges, explore them.
- Mark vertices to avoid getting stuck in a loop.

Both take time \(O(V+E)\)