Example Directed Graph (Digraph)

V = \{a,b,c,d,e,f\}
E = \{(a,b), (a,c), (a,e), (b,c), (b,d), (b,e), (c,d),
      (c,f), (d,e), (d,f), (e,f)\}

|V| = 6, |E| = 11

Example Undirected Graph

An undirected graph is just like a directed graph, except the edges are unordered pairs (sets) \{(u,v)\}

Example:

Social Network Graph

Graph Concepts and Algorithms

- Adjacency Matrix
- Topological Sort
- Search
  - depth-first search
  - breadth-first search
- Shortest paths
  - Dijkstra's algorithm
- Minimum spanning trees
  - Prim's algorithm
  - Kruskal's algorithm

Others:
- Graph Coloring
- Planarity
- Traveling Salesman problem.

Graph Adjacency Matrix

<table>
<thead>
<tr>
<th></th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>4</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>
Graph Adjacency Matrix

Squaring adjacency matrix gives number of possible paths of length 2!

Cubing adjacency matrix gives number of possible paths of length 3!

Same idea holds for undirected graphs.

How would you find friends of friends?

Find the number of common friends?

Graph Concepts and Algorithms

- Adjacency Matrix
- Topological Sort
- Search
  - depth-first search
  - breadth-first search
- Shortest paths
  - Dijkstra’s algorithm
- Minimum spanning trees
  - Prim’s algorithm
  - Kruskal’s algorithm

Others:
- Graph Coloring
- Planarity
- Traveling Salesman problem.
Search if a person is someone’s Friend?

Graph Concepts and Algorithms

- Adjacency Matrix
- Topological Sort
- Search
  - depth-first search
  - breadth-first search
- Shortest paths
  - Dijkstra’s algorithm
- Minimum spanning trees
  - Prim’s algorithm
  - Kruskal’s algorithm

Others:
- Graph Coloring
- Planarity
- Traveling Salesman problem.

Shortest Path Algorithm: Used Everyday!

Train Network in France.

Graph Concepts and Algorithms

- Adjacency Matrix
- Topological Sort
- Search
  - depth-first search
  - breadth-first search
- Shortest paths
  - Dijkstra’s algorithm
- Minimum spanning trees
  - Prim’s algorithm
  - Kruskal’s algorithm

Others:
- Graph Coloring
- Planarity
- Traveling Salesman problem.

Shortest Path Algorithm: Used Everyday!

Maps! GPS directions!

Topological Sort

- Topological sort of the dag
  This is a numbering of the vertices such that all edges go from lower- to higher-numbered vertices

- Useful in job scheduling with precedence constraints
Tell Me Dave: Making ‘Avoffato’

Topological Sort
- **Topological sort** of the dag
  - This is a numbering of the vertices such that all edges go from lower- to higher-numbered vertices
  - 1 Move Bowl to coffee machine
  - 2 Turn coffee machine tap
  - 3 Scoop Ice-cream
  - 4 Put ice-cream in bowl
  - 5 Useful in job scheduling with precedence constraints

Graph Concepts and Algorithms
- **Adjacency Matrix**
- **Topological Sort**
- **Search**
  - depth-first search
  - breadth-first search
- **Shortest paths**
  - Dijkstra’s algorithm
  - Minimum spanning trees
    - Prim’s algorithm
    - Kruskal’s algorithm
- Others:
  - Graph Coloring
  - Planarity
  - Traveling Salesman problem.

AI: Vision and Robotics
- Number of cell-phones in 2014: 7 billion!
- So many images everywhere.
- But do computers understand the images?

Without Graph Algorithms
Undirected Graph

Modeling the Activities with an Undirected Graph

Robot Sees the Future!

Anticipating Future Actions

Robot’s view

Humans use anticipation all the time
e.g., interacting with other people, playing sports, driving, etc.

Future anticipations

Future anticipations

Graph Concepts and Algorithms

- Adjacency Matrix
- Topological Sort
- Search
  - depth-first search
  - breadth-first search
- Shortest paths
  - Dijkstra’s algorithm
- Minimum spanning trees
  - Prim’s algorithm
  - Kruskal’s algorithm

Others:
- Graph Coloring
- Planarity
- Traveling Salesman problem.
In 2012, we started a company called Cognical.

Problem:
65 million US consumers cannot finance electronics, furniture, appliances and other durable goods online.

Cognical collects data about customers from various sources.

Reveals over a graph, and predicts how to finance.

Graphs in Practice

65 million US consumers have low FICO scores but are broadly creditworthy.

- Browse Online
  - but lack a payment option to buy
- Buy at local stores
  - with cash or finance via Lease to Own (LTO)

Graphs:

- Adjacency Matrix
- Topological Sort
- Search
  - depth-first search
  - breadth-first search
- Shortest paths
  - Dijkstra’s algorithm
- Minimum spanning trees
  - Prim’s algorithm
  - Kruskal’s algorithm

Others:

- Graph Coloring
- Planarity
- Traveling Salesman problem.