Graphs

c++,graphs,lecture16

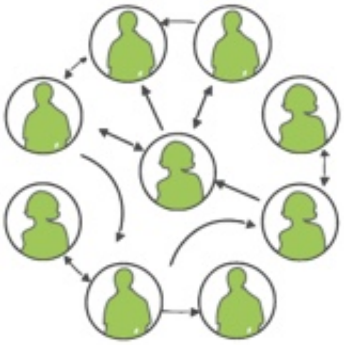
- an ADT that stores a set of **entities** and keeps track of **relationships** between all of them
    - similar to how Core Data works in iOS

Entity/Relationship Examples

- People
    - Joe is friends with Linda
- Cities
    - LA is 3000 miles from NYC
- Web pages
    - ucla.edu links to awesome.com

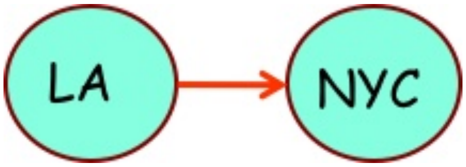
Each graph holds two types of items:

- **vertices (aka nodes)**
- **edges(aka arcs)**



Directed vs. Undirected Graphs

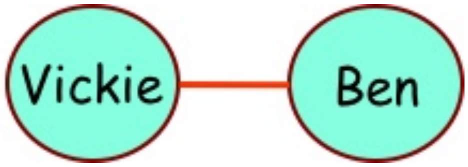
Directed Graph

- an edge goes from one vertex to another in a **specific direction**
- 

- e.g. a flight from LA to NYC but not the other way around

Undirected Graph

- all edges are **bi-directional**

- 

- e.g. Facebook friends

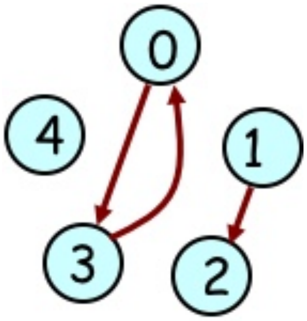
Representing a Graph

Adjacency Matrix

- each element in the array indicates whether or not there is an edge between vertex **i** and vertex **j**
- nodes could point to themselves

```
bool graph[5][5];
graph[0][3] = true;
graph[1][2] = true;
graph[3][0] = true;
```

	0	1	2	3	4
0				T	
1			T		
2					
3	T				
4					



- to bi-directionally connect vertices i and j:
  - `array[i][j] = true;`
  - `array[j][i] = true;`

Traveling Salesman Problem

What is the big-O of finding cheapest route to travel through all N cities through an airline?

N!