

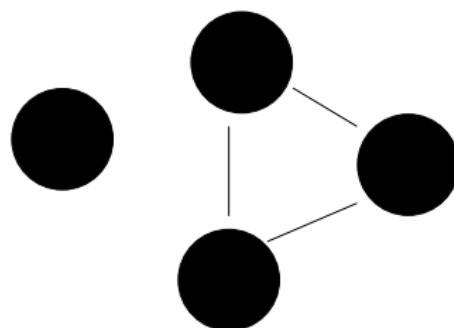
# Graph Theory: Introduction

David Jacobo  
[jguillen@cimat.mx](mailto:jguillen@cimat.mx)

September 11, 2015

# Definition

$$G = (V, E)$$



# Outline

## 1 Motivation

- Seven Bridges of Königsberg
- Six degrees of separation theory
- More problems

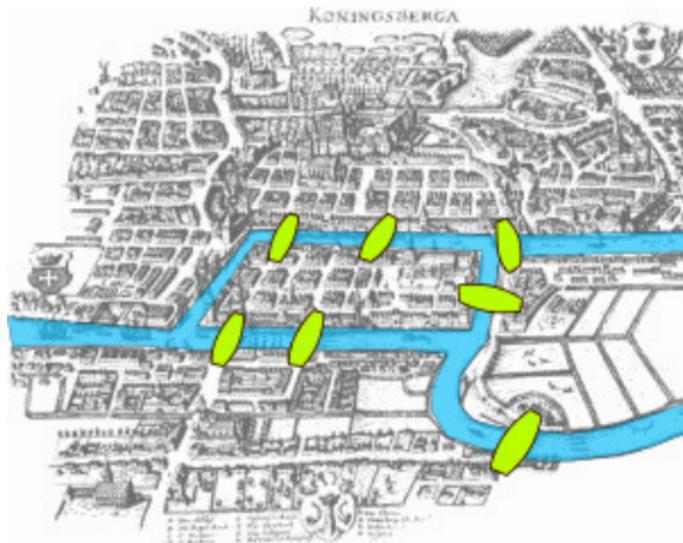
## 2 Graph theory basics

- Terminology
- Graph representation

## 3 Graph traversal

- Deep First Search (DFS)
- Breadth First Search (BFS)

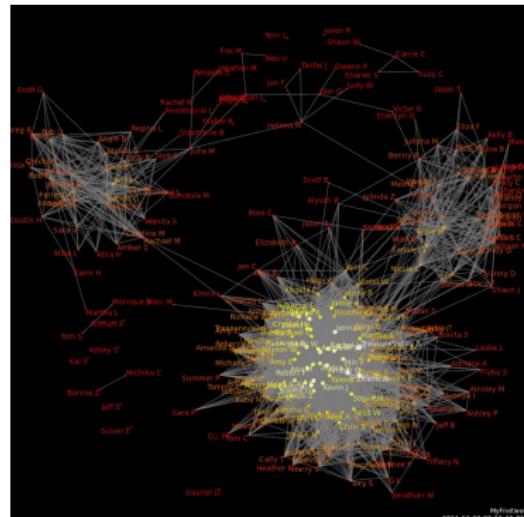
# Seven Bridges of Königsberg



"Okay" - Leonhard Euler (not really)

# Six degrees of separation theory

"Six degrees of separation is the theory that everyone and everything is six or fewer steps away, by way of introduction, from any other person in the world ... It was originally set out by Frigyes Karinthy in 1929..." - Wikipedia



# Daily applications

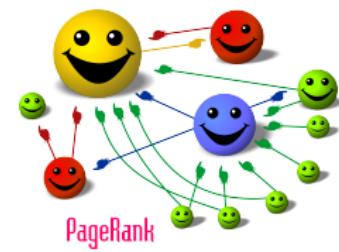
Traffic



Traveling



Ranking



# Outline

## 1 Motivation

- Seven Bridges of Königsberg
- Six degrees of separation theory
- More problems

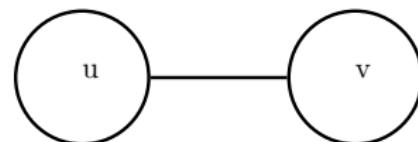
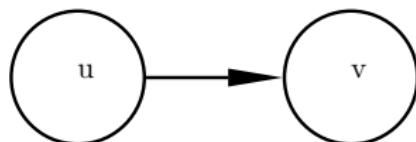
## 2 Graph theory basics

- Terminology
- Graph representation

## 3 Graph traversal

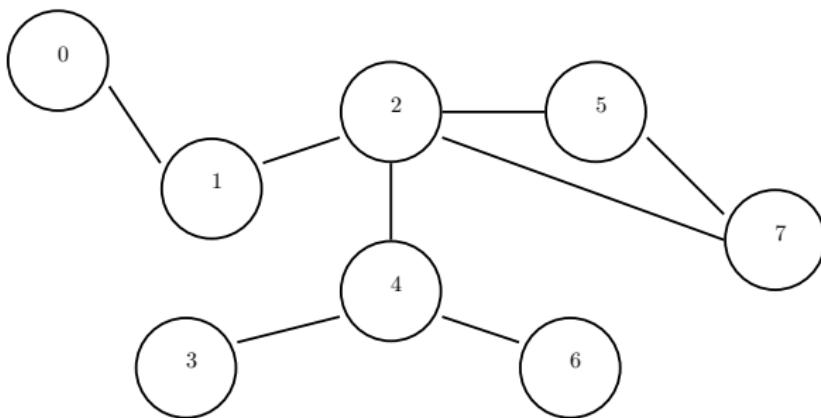
- Deep First Search (DFS)
- Breadth First Search (BFS)

# Directed vs Undirected



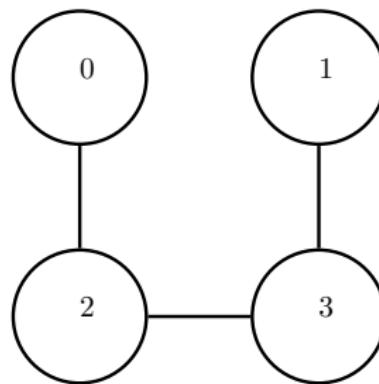
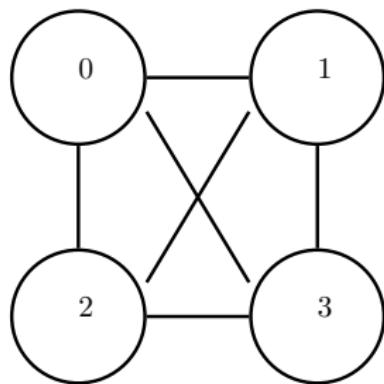
For a directed graph,  $u \rightarrow v$  doesn't imply  $v \rightarrow u$ .

# Paths and cycles



A path is a succession of distinct vertices which are connected.  
A cycle is also a path, in which the first and the last vertex are the same.

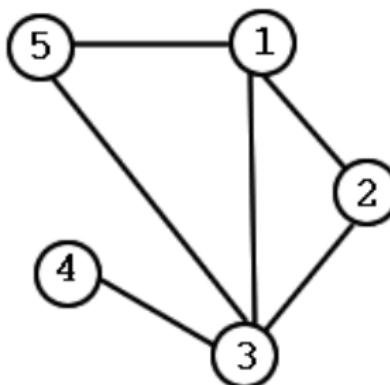
# Dense vs Sparse



Leftmost figure is known as a **clique** and rightmost is a **tree**

# Adjacency Matrix

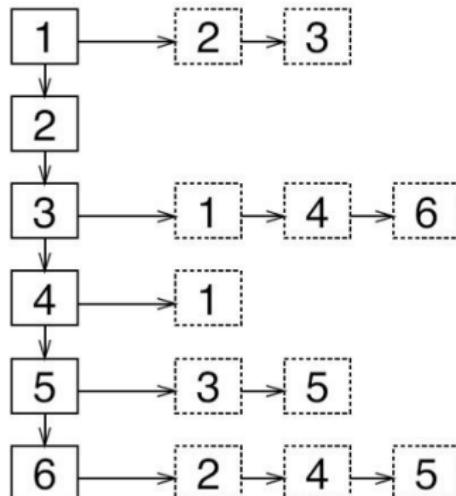
Having a **VxV matrix**, the space complexity for this is  **$O(V^2)$** . Preferred for dense graphs or for using with algorithms which constantly require to check if a particular edge exists.



$$\Rightarrow A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 \end{bmatrix}$$

# Adjacency List

List which contains the direct neighbours per each of the vertices, the memory space usage is  $O(V + E)$ . Preferred when our graph is sparse or the algorithm need to list the outgoing edges efficiently.



# Outline

## 1 Motivation

- Seven Bridges of Königsberg
- Six degrees of separation theory
- More problems

## 2 Graph theory basics

- Terminology
- Graph representation

## 3 Graph traversal

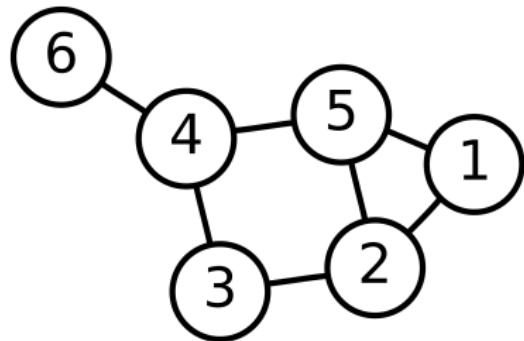
- Deep First Search (DFS)
- Breadth First Search (BFS)

# DFS Algorithm

Pre-requisite

```
1   for each node n in G:  
2       n.parent = NIL
```

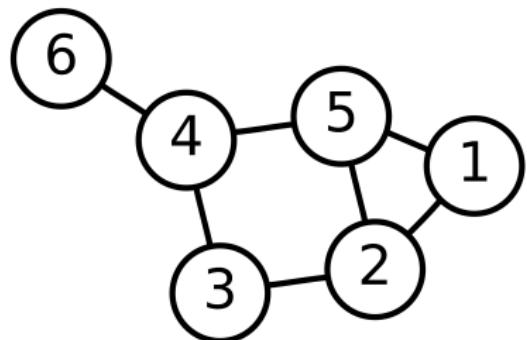
```
1 DFS(G,v):  
2     label v as discovered  
3     for all valid edges v -> w do  
4         if vertex w !discovered then  
5             w.parent = v  
6             recursively call DFS(G,w)
```



Time complexity: **O(V+E)**

# BFS Algorithm

```
1 Breadth-First-Search(G, v):
2
3     for each node n in G:
4         n.distance = INFINITY
5         n.parent = NIL
6
7     create empty queue Q
8
9     v.distance = 0
10    Q.enqueue(v)
11
12    while Q is not empty:
13
14        u = Q.dequeue()
15
16        for each node n that is adjacent to u:
17            if n.distance == INFINITY:
18                n.distance = u.distance + 1
19                n.parent = u
20                Q.enqueue(n)
```



Time complexity: **O(V+E)**

# Q & A



## References

- Competitive Programming site
- Algorists' repository