Graphs

Anders Kalhauge and Marjahan Begum

Graphs

- Set of vertices connected pairwise by edges.
- Collection of nodes/vertices connected to each other through set of edges
- Edges connected in any possible way
- G = (V,E)

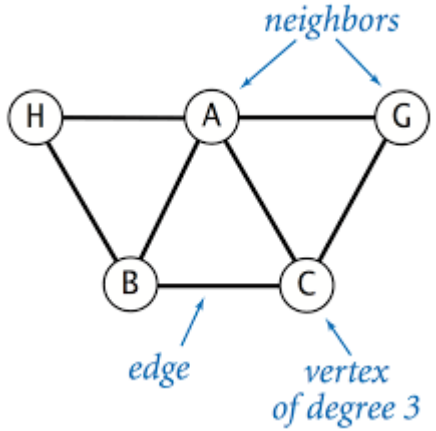

Zealand
Øresund
Roskilde Fjord
Amager
Køge Bugt
Nordbanen
Klampenborg-banen
Hareskovbanen
Frederikssundbanen
Ring-banen
Vestbanen
Køgebugtbanen
Hillerød
Allerød
Birkerød
Holte
Virum
Sorgenfri
Lyngby
Jægersborg
Gentofte
Bernstorffsvej
Klampenborg
Ordrup
Charlottenlund
Farum
Værløse
Hareskov
Skovbrynet
Bagsværd
Stengården
Buddinge
Kildebakke
Vangede
Dyssegård
Emdrup
Ryparken
Hellerup
Svanemøllen
Nordhavn
Østerport
Nørreport
Vesterport
København H
Copenhagen Central Station
Frederikssund
Ølstykke
Egedal
Stenløse
Veksø
Kildedal
Måløv
Ballerup
Malmparken
Skovlunde
Herlev
Husum
Islev
Jyllingevej
Vanløse
Flintholm
Bispebjerg
Nørrebro
Fuglebakken
Grendal
Peter Bangs Vej
Langgade
Enghave
Dybbølsbro
Valby
KB Hallen
Ålholm
Høje Taastrup
Taastrup
Albertslund
Glostrup
Brøndbyøster
Rødovre
Hvidovre
Danshøj
Vigerslev Allé
Sydhavn
Sjælør
Ny Ellebjerg
Åmarken
Frihedsn
Avedøre
Brøndby Strand
Vallensbæk
Ishøj
Hundige
Greve
Karlslunde
Solrød Strand
Jersie
Ølby
Køge
Copenhagen
S-train Diagram
Symbols at stations
service always stops here
service never stops here
when more than one service stops, you are always able to transfer
(this is highlighted with a grey border on lesser obvious stations)
occasional terminus

Definitions

- **Path.** Sequence of vertices connected by edges.
- **Cycle**. Path whose first and last vertices are the same.
- Two vertices are **connected** if there is a path between them.

A simple graph

Vertices	Neighbours
A	B C G H
B	A C H
C	A B G
G	A C
H	A B



Graph types

- Acyclic graph has no cycles
  - A tree – an acyclic connected graph

- Forest – disjoint set of trees

Tree

- V-1 edges and no cycles
- V-1 edges and is connected
- Connected and moving any edge disconnects it
- Acyclic but adding any edge creates a cycles
- Exactly one simple plath connects each pair of vertices in G

Applications of Graphs

- Highway network, Flight network.
- Computer networks: Local area network, Internet, Web.
- Representing relationships between components in electronic circuits.
- Facebook Graph Search is a real-life example of application of graph algorithms for example friends suggestions

Applications of Graphs

- Biology and conservation efforts where a vertex represents regions where certain species exist and the edges represent migration path or movement between the regions.

- Google Maps: Various locations are represented as vertices and the roads are represented as edges and graph theory is used to find shortest path between two nodes.

- E-commerce The “Recommendations for you” section on various e-commerce websites uses graph theory to recommend items of similar type to user’s choice.

Graph continues

- Directed
- Undirected
- Weighted and unweighted

Graph Representation

- Edge List representation
- Operations
  - Finding nodes adjacent to a given nodes O(|E|)
  - Finding if two node are connected O(|E|)
  - Number of edges can be very large
  - O(|V|) better?

Adjacency Matrix

0	**1**	**2**	**3**	**4**	**5**
1					
2					
3					
4					
5					

Time Complexity

- Findng adjacent vertices
  - Go to the vertices list O (v)
  - Then go to the row associated with the vertice in the two dimensional array O(v)
  - Total O(v)
- Finding if two nodes are connected
  - O(1) if you know i and j indexes
  - O(v + v + 1) = O(v)

Time Complexity

- Finding if two nodes are connected
  - O(1) + O(V)
  - How can the second O can be avoided?

- How would you represent weighted graph?

Memory Big O?

- V^2
- Sparse vs dense graph
- Redundant information when it is not connected
- Most graph will not be anything close to v^2
- Adjacency Matrix will not be a good fit

What is the solution

- Only keep listed of nodes that are connected using indexes
- List
  - Array v1 = new int[?]
  - Linked list
  - Binary tree
- Space O(e)
- $E < V^2$

Memory usage

- 2 x e (edges) (undirected graph)
- Exactly e edges (directed graph)
- Space is proportional to number of edges

Time Complexity

- Two nodes connected?
  - O(1) Aij in adjacency matrix
  - Linear search – O(v)
    - Sort the binary search?
      - O(log v)
- Adjacent nodes
  - O(v) for both structure

Simple Exercise

- Design a social network of friends in the class using a graph data structure
- Get inspirations from
  - http://introcs.cs.princeton.edu/java/45graph/Graph.java
  - https://www.cs.duke.edu/courses/cps100e/fall10/class/11_Bacon/code/Graph.html