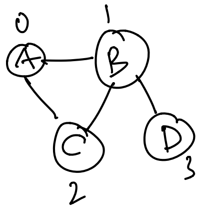
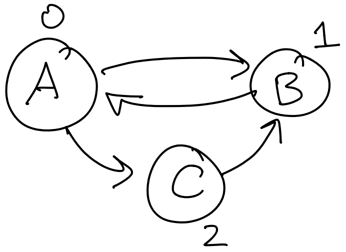
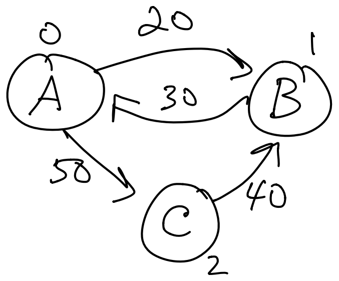
- Directed / Undirected
- Weighted / Unweighted







Adjacency matrix OR Adjacency list

• Number all the vertices in the graph.
0, 1, 2, 3...

	0	1	2
0	0	1	1
1	1	0	0
2	0	1	0

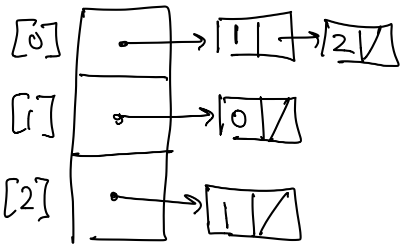
(unweighted)

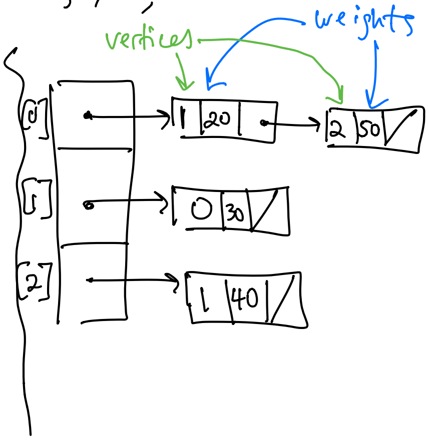
Example
A → B
0 → 1
B → A
1 → 0

	0	1	2
0	0	20	50
1	30	0	0
2	0	40	0

(weighted)

• Number the ~~ve~~ vertices 0, 1, 2, 3....





Key	Values	Edges
A	B, C	
B	A	
C	B	
D	B, 10	

Add new edge
D → B
weight = 10

Key	Value
Vertex	Population

Extra Info

Real World

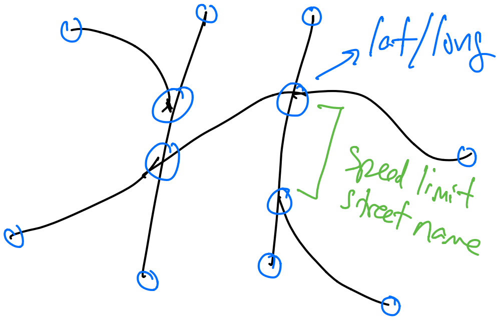
Collection of latitude/longitude coordinates → has a unique integer ID

Speed limit between each lat/long coordinate

Street name " " " "

Intersections : Vertex

Streets : Edge



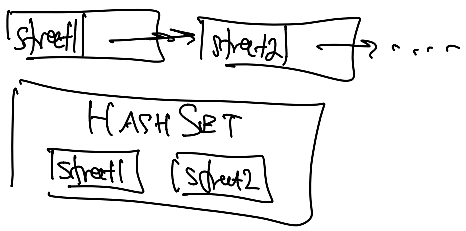
Map

Key: Lat/long coord →

Key: unique int ID →

Values (Set/List)

Streets



Street class

{
Vertex1, Vertex2
Street name
street speed limit
}

2nd Map

Unique ID	→	Lat/Long
Key		Value

ADTs		Impls
List	→	ArrayLists Linked Lists (singly/doubly)
Sorted List	→	Array List Linked List
Stack Queue	→	Arrays/linked lists
Set Maps	→	BST Hashtables
Priority Queue	→	Heap
Graph	→	Adjacency lists " matrices

Sorting

$O(n^2)$ { Insertion, Selection, Bubble

$O(n\log n)$ { Quick, Merge