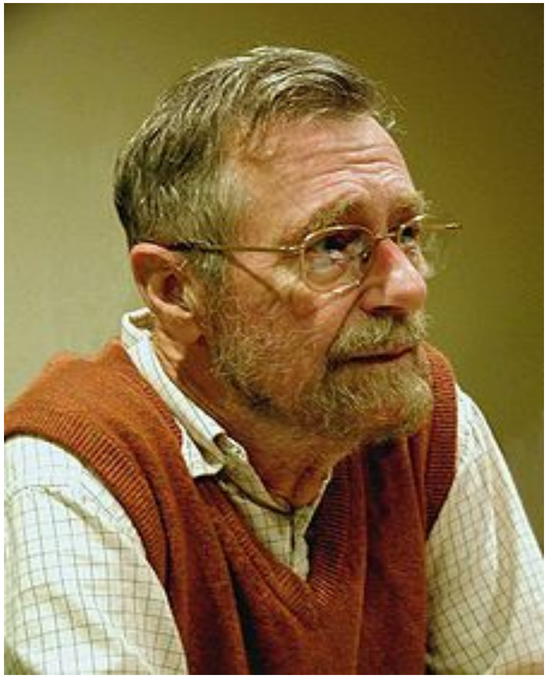
A presentation on

“ DIJKSTRA ALGORITHM”

The author: Edsger Wybe Dijkstra



- May 11, 1930 – August 6, 2002
- Received the 1972 A. M. Turing Award, widely considered the most prestigious award in computer science.
- The Schlumberger Centennial Chair of Computer Sciences at The University of Texas at Austin from 1984 until 2000
- Made a strong case against use of the GOTO statement in programming languages and helped lead to its deprecation.
- Known for his many essays on programming.

Dijkstra's algorithm

Problem Definition: From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.

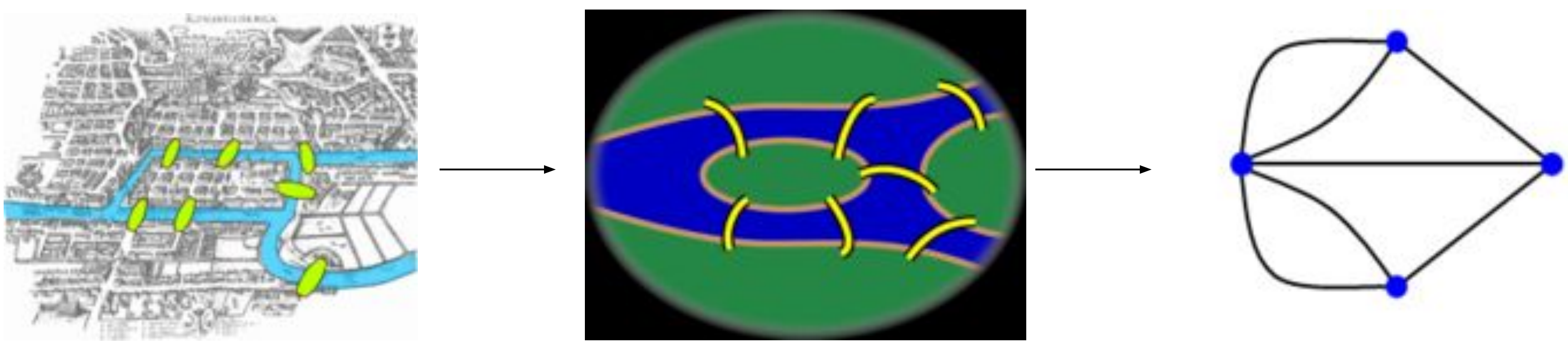
CSE@GIT

Objectives of the Experiment:

1. To apply the Greedy Technique.
2. Present the working of Dijkstra Algorithm.
3. Learn to write Algorithm in standard form.
4. Find out shortest path from single source to other nodes in the graph

Graph Theory - History

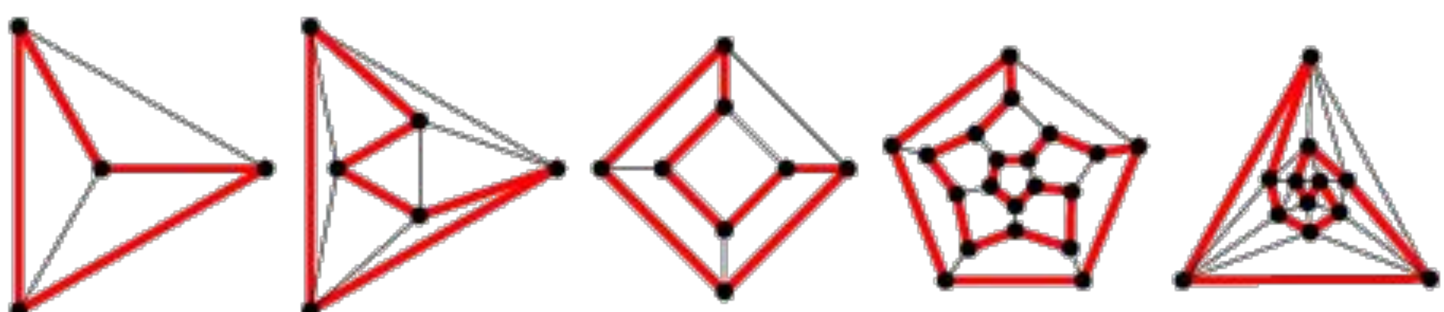
Leonhard Euler's paper on *"Seven Bridges of Königsberg"* , published in 1736.



Graph Theory - History

Cycles in Polyhedra

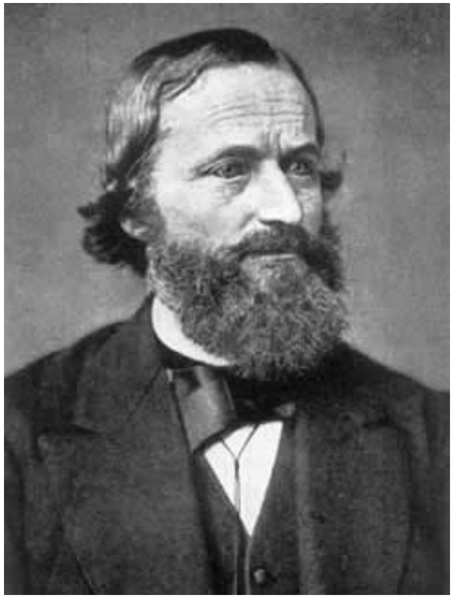
Thomas P. Kirkman William R. Hamilton



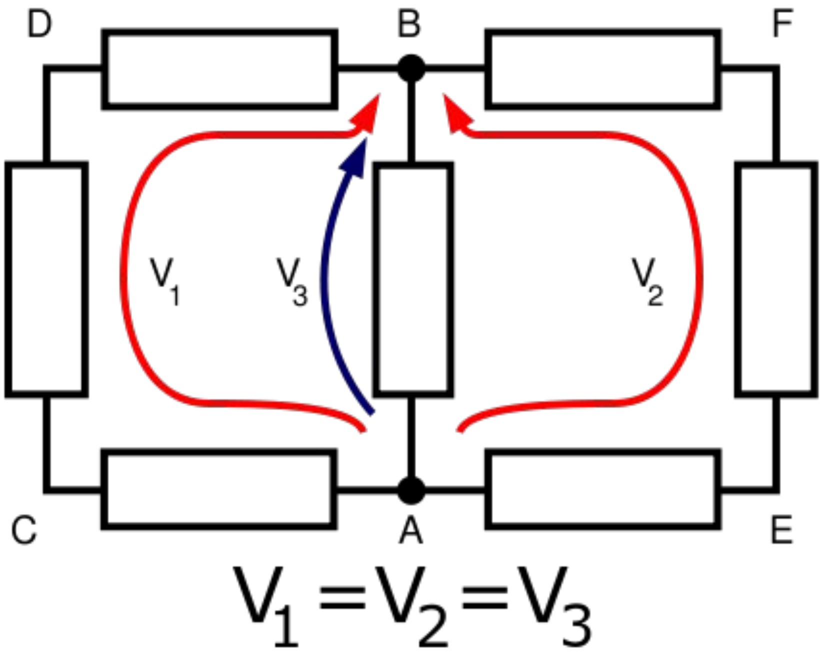
Hamiltonian cycles in Platonic graphs

Graph Theory - History

Trees in Electric Circuits

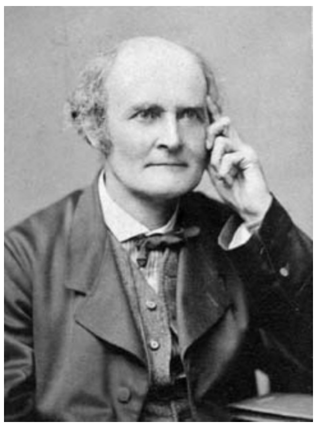


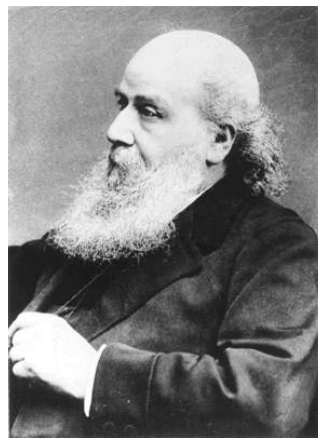
Gustav Kirchhoff



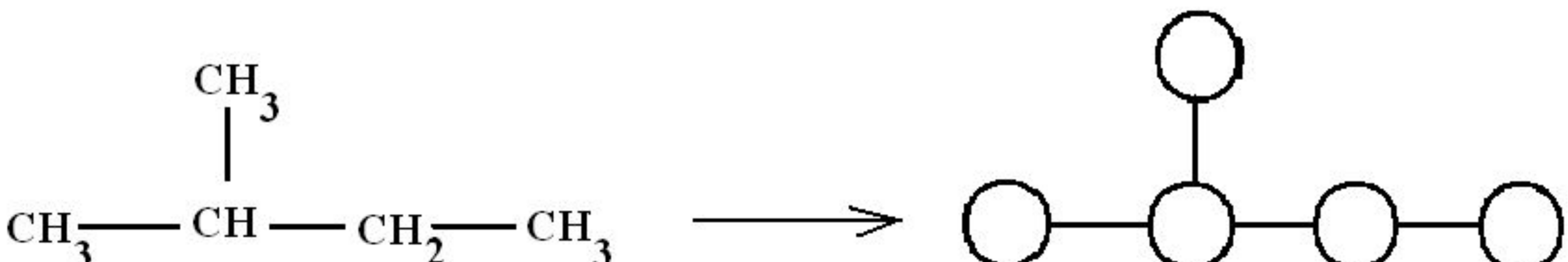
Graph Theory - History

Enumeration of Chemical Isomers



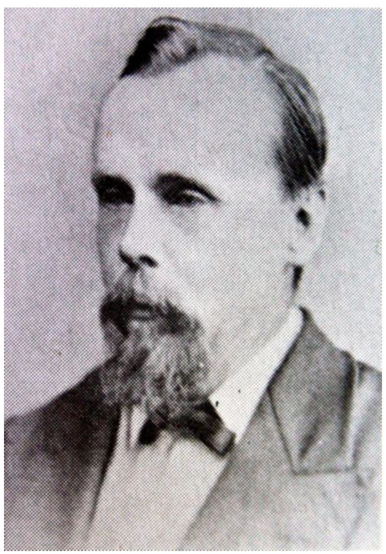


Arthur Cayley James J. Sylvester George Polya

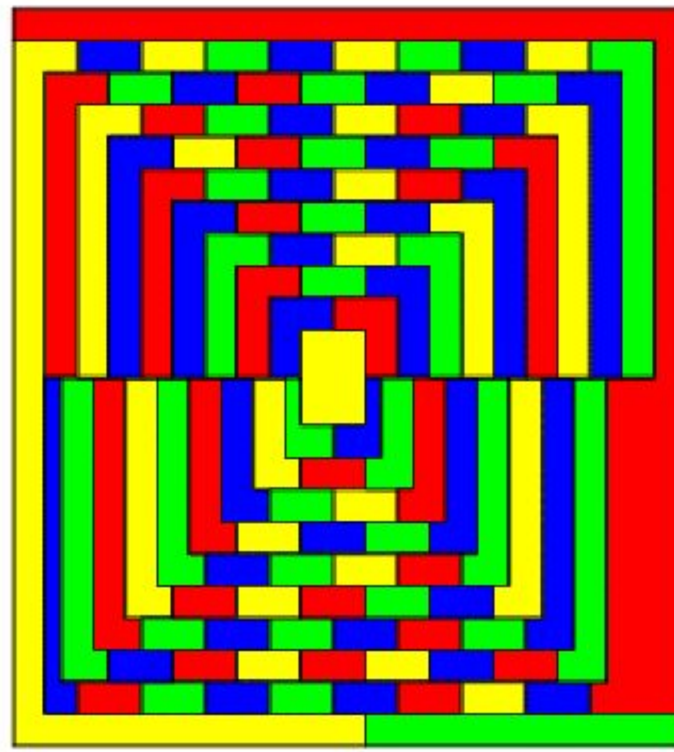


Graph Theory - History

Four Colors of Maps



Francis Guthrie Auguste DeMorgan



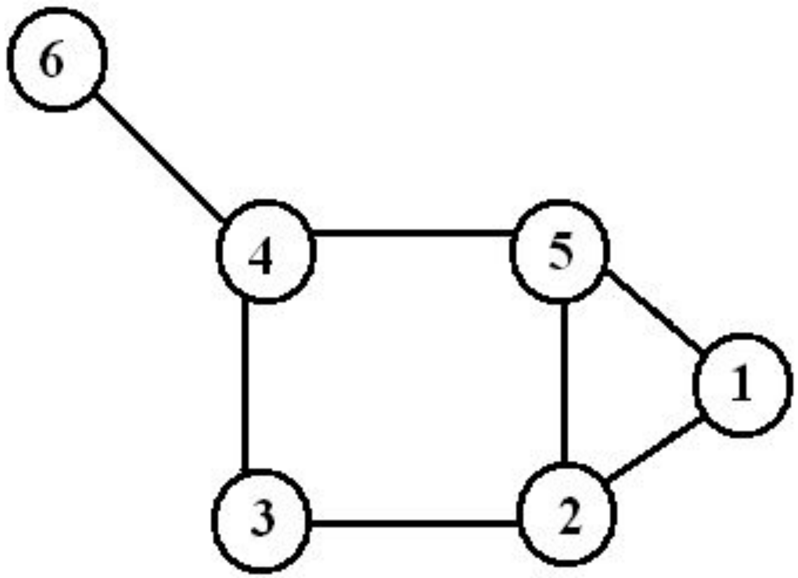
Definition: Graph

- G is an ordered triple G:=(V, E, f)
  - V is a set of nodes, points, or vertices.
  - E is a set, whose elements are known as edges or lines.
  - f is a function
    - maps each element of E
    - to an unordered pair of vertices in V.

Definitions

- Vertex
  - Basic Element
  - Drawn as a *node* or a *dot*.
  - V**ertex set** of G is usually denoted by $V(G)$, or V
- Edge
  - A set of two elements
  - Drawn as a line connecting two vertices, called end vertices, or endpoints.
  - The edge set of G is usually denoted by E(G), or E.

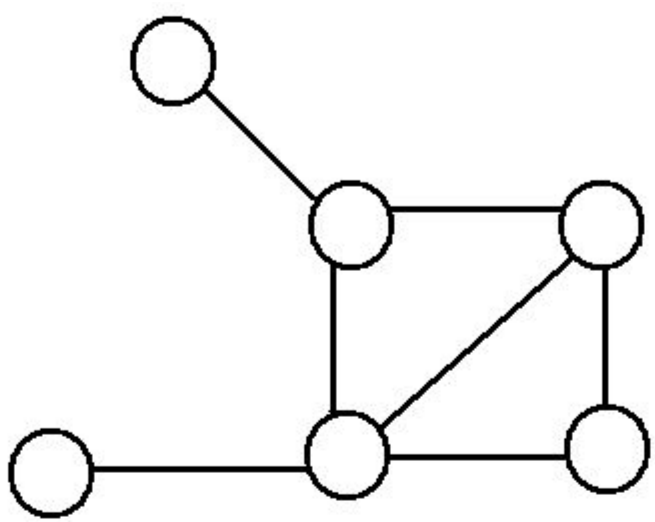
Example



- V:={1,2,3,4,5,6}
- E:={{1,2},{1,5},{2,3},{2,5},{3,4},{4,5},{4,6}}

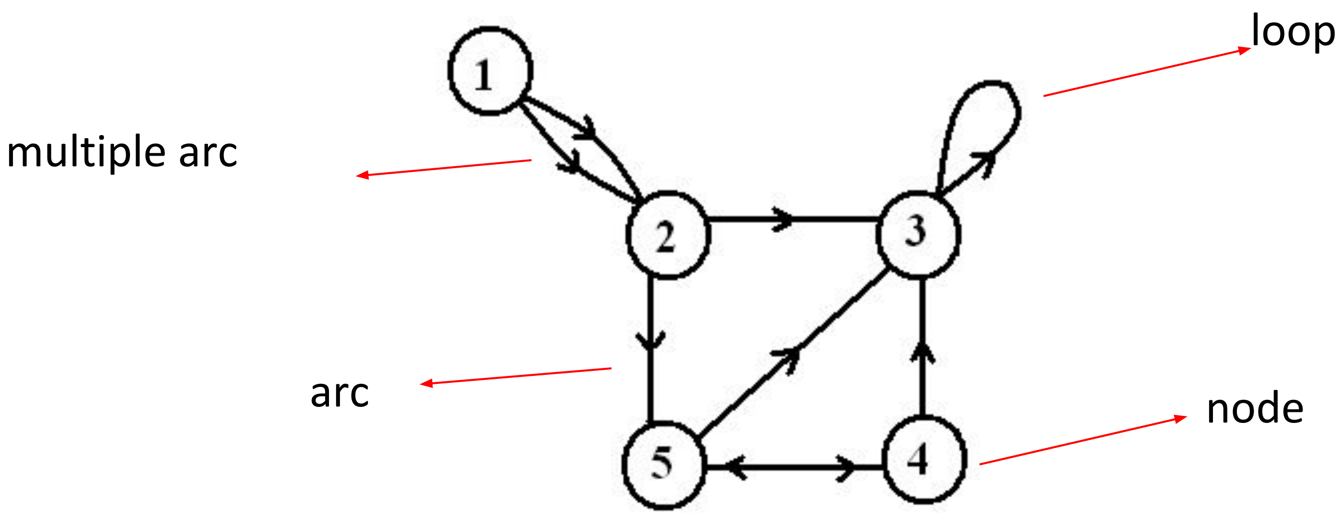
Simple Graphs

Simple graphs are graphs without multiple edges or self-loops.



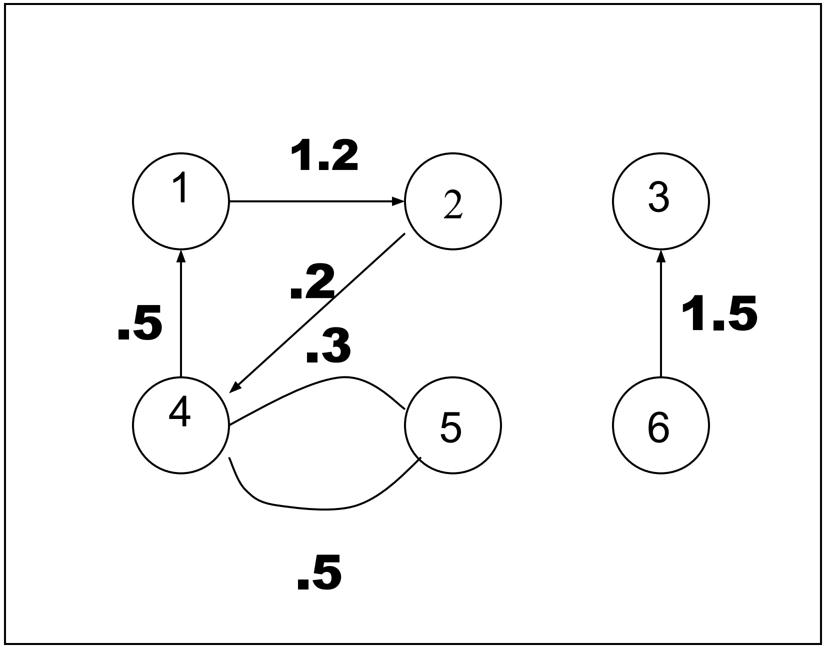
Directed Graph (digraph)

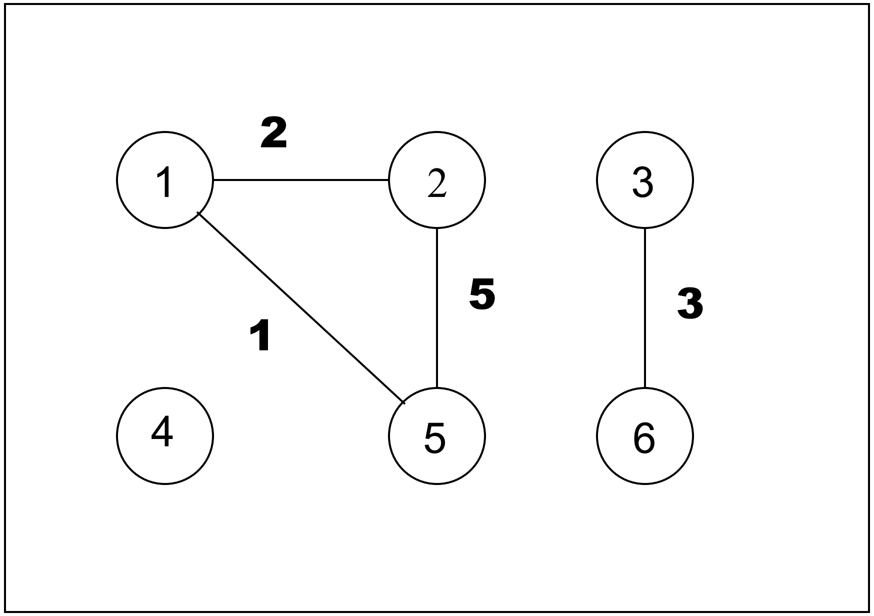
- Edges have directions
  - An edge is an *ordered* pair of nodes



Weighted graphs

- is a graph for which each edge has an associated ***weight***, usually given by a ***weight function*** $w: E \rightarrow \mathbf{R}$.





Greedy Technique

Constructs a solution to an optimization problem piece by piece through a sequence of choices that are:

Defined by an objective function and a set of constraints

- feasible, i.e. satisfying the constraints
- locally optimal (with respect to some neighborhood definition)
- greedy (in terms of some measure), and irrevocable

For some problems, it yields a globally optimal solution for every instance. For most, does not but can be useful for fast approximations. We are mostly interested in the former case in this class.

Applications of the Greedy Strategy

- Optimal solutions:
  - change making for “normal” coin denominations
  - minimum spanning tree (MST)
  - single-source shortest paths
  - simple scheduling problems
  - Huffman codes
- Approximations/heuristics:
  - traveling salesman problem (TSP)
  - knapsack problem
  - other combinatorial optimization problems

Animations of Dijkstra's algorithm

- Animation -1
- Animation -2
- Animation -3

Shortest paths – Dijkstra's algorithm

Single Source Shortest Paths Problem: Given a weighted connected (directed) graph G, find shortest paths from source vertex s to each of the other vertices

Dijkstra's algorithm: Similar to Prim's MST algorithm, with a different way of computing numerical labels: Among vertices not already in the tree, it finds vertex u with the smallest sum

$$d_v + w(v,u)$$

where

v is a vertex for which shortest path has been already found on preceding iterations (such vertices form a tree rooted at s)

d_v is the length of the shortest path from source s to v

$w(v,u)$ is the length (weight) of edge from v to u

Dijkstra’s Algorithm

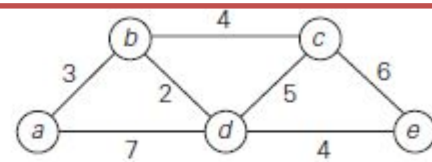
ALGORITHM *Dijkstra(G, s)*

//Dijkstra’s algorithm for single-source shortest paths
//Input: A weighted connected graph $G = \langle V, E \rangle$ with nonnegative weights
// and its vertex s
//Output: The length d_v of a shortest path from s to v
// and its penultimate vertex p_v for every vertex v in V

```
Initialize(Q)   //initialize priority queue to empty
for every vertex v in V
    d_v ← ∞;   p_v ← null
    Insert(Q, v, d_v)   //initialize vertex priority in the priority queue
d_s ← 0;   Decrease(Q, s, d_s)   //update priority of s with d_s
V_T ← ∅
for i ← 0 to |V| − 1 do
    u* ← DeleteMin(Q)   //delete the minimum priority element
    V_T ← V_T ∪ {u*}
    for every vertex u in V − V_T that is adjacent to u* do
        if d_u* + w(u*, u) < d_u
            d_u ← d_u* + w(u*, u);   p_u ← u*
            Decrease(Q, u, d_u)
```

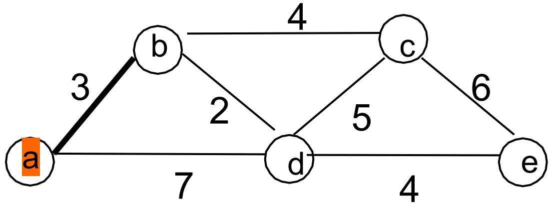
Intitial call is dijkstra(G[][], V)

Example -1

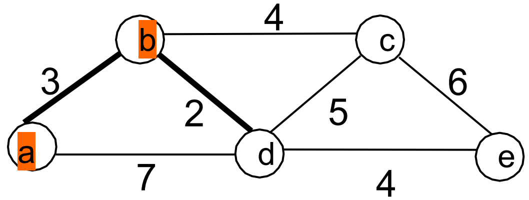


Tree vertices **Remaining vertices**

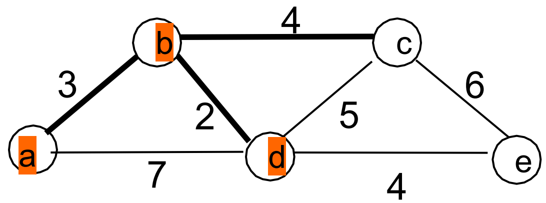
a(-,0) b(a,3) c(-,∞) d(a,7) e(-,∞)



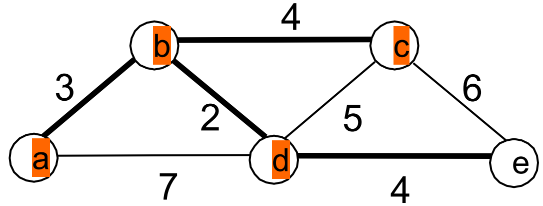
b(a,3) c(b,3+4) d(b,3+2) e(-,∞)



d(b,5) c(b,7) e(d,5+4)



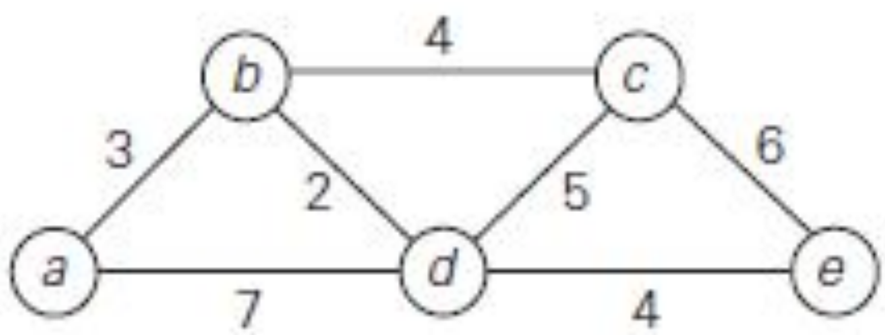
c(b,7) e(d,9)



e(d,9)

Example -1

- from *a to b : a – b of length 3*
- from *a to d : a – b – d of length 5*
- from *a to c : a – b – c of length 7*
- from *a to e : a – b – d – e of length 9*



Example -1 execution

Enter the number of vertices

5

Enter the Weighted Matrix for the graph

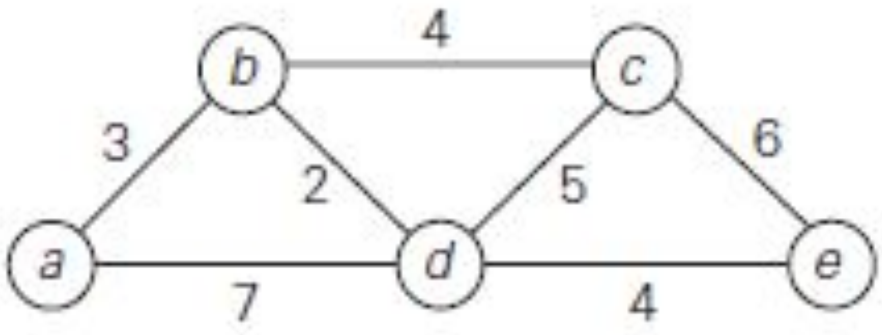
0 3 0 7 0

3 0 4 2 0

0 4 0 5 6

7 2 5 0 4

0 0 6 4 0



Enter the source

1

The Shorted Path to all nodes are

1 to 1 is 0

1 to 2 is 3

1 to 3 is 7

1 to 4 is 5

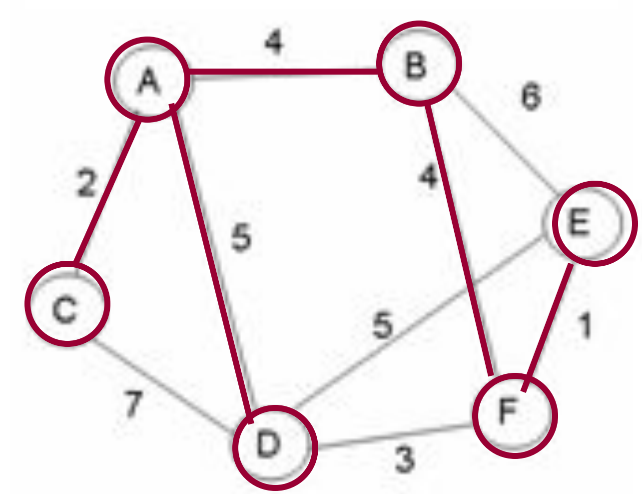
1 to 5 is 9

BUILD SUCCESSFUL (total time: 51 seconds)

Example -2

Node	Included	Distance	Path
A	t	-	-
B	~~f~~ t	4	A
C	~~f~~ t	2	A
D	~~f~~ t	5	A
E	~~f~~ t	~~∞~~ ~~10~~ 9	~~-~~ ~~B~~ F
F	~~f~~ t	~~∞~~ 8	~~-~~ B

Give the shortest path tree for node A for this graph using Dijkstra's shortest path algorithm. Show your work with the 3 arrays given and draw the resultant shortest path tree with edge weights included.



Efficiency of Dijkstra's algorithm

- Correctness can be proven by induction on the number of vertices.
- Doesn't work for graphs with negative weights (whereas Floyd's algorithm does, as long as there is no negative cycle). Can you find a counterexample for Dijkstra's algorithm?
- Applicable to both undirected and directed graphs
- Efficiency
  - $O(|V|^2)$ for graphs represented by weight matrix and array implementation of priority queue
  - $O(|E|\log|V|)$ for graphs represented by adj. lists and min-heap implementation of priority queue

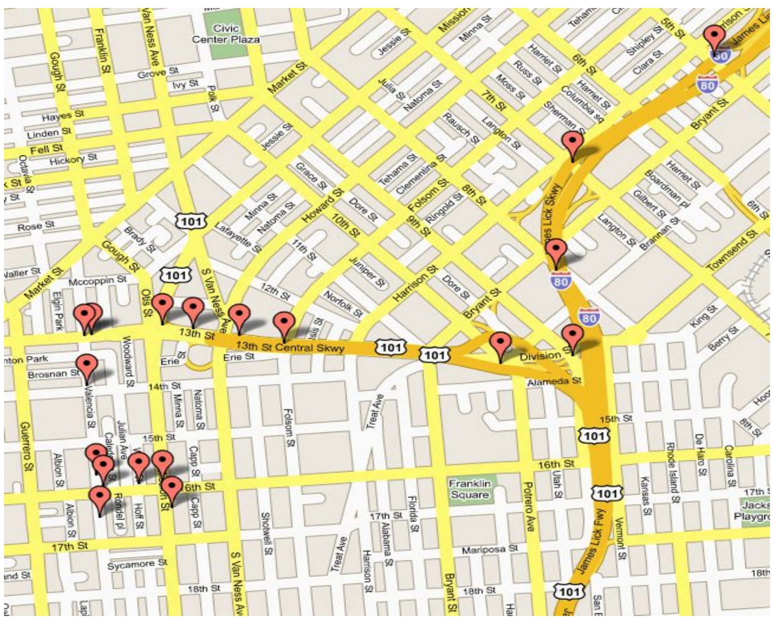
3D Animation of Dijkstra’s algorithm

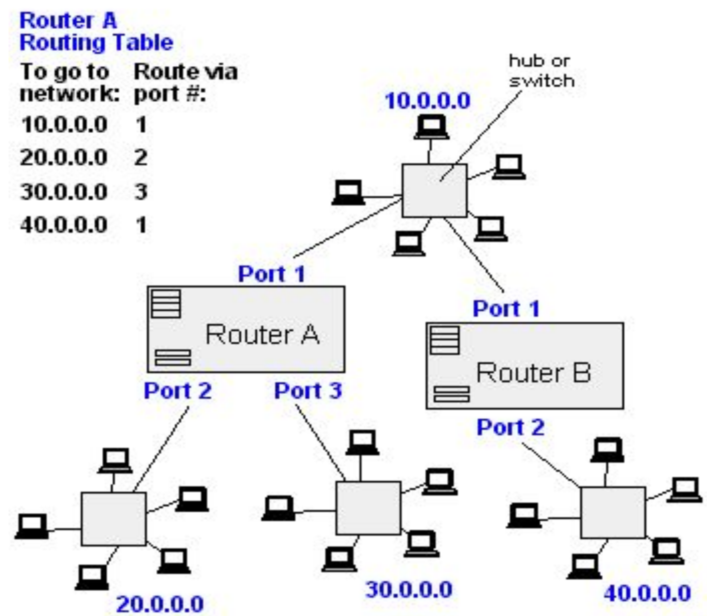
- Animation -3

Applications of Dijkstra's Algorithm

- Traffic Information Systems are most prominent use
- Mapping (Map Quest, Google Maps)
- Routing Systems
- Prof. Lauren Meyers (Biology Dept.) uses networks to model the spread of infectious diseases and design prevention and response strategies.

From Computer Desktop Encyclopedia
© 1998 The Computer Language Co. Inc.





Learning Outcome of the Experiment and Conclusion

At the end of the session, students should be able to :

1. Explain the working of Greedy technique. [L2, CO 2, PO1]
2. Demonstrate the working of Dijkstra algorithm. [L3, CO 2, PO3]
3. Compute the shortest path between a source and all vertices in a weighted connected graph. [L3, CO 2, PO3]