

Lecture 4: Graphs, graph representation, basic graph processing

BE5B33ALG — Algorithms

Ing. Robert Pěnička, Ph.D.
doc. RNDr. Daniel Průša, Ph.D.

Faculty of Electrical Engineering
Czech Technical University in Prague



FACULTY
OF ELECTRICAL
ENGINEERING
CTU IN PRAGUE



MULTI-ROBOT
SYSTEMS
GROUP

Introduction

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

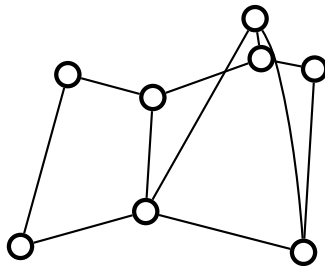
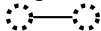
Graphs

Graph representations

- Graph $G = (V, E)$
- V is a set of nodes (vertices)
- E is a set of edges (connections between nodes)
- Node (Vertex)



- Edge



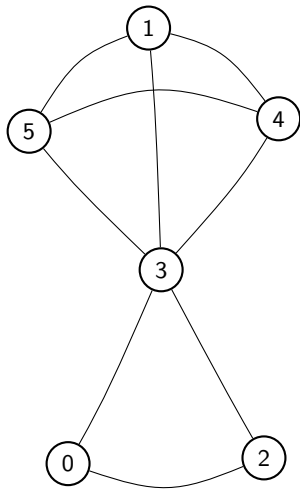
Graphs: Nodes and Edges

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations



□ Nodes, Vertices

- Servers, cities...
- Persons, people...
- Objects in comp. science
- ... etc.

□ Edges

- Connections, roads...
- Personal relations
- Relations among objects
- ... etc.

Usual graph representations

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations

Node degrees & Lists of neighbours

- Less obvious, more effective
- 1D/2D array, vector, ArrayList...

Node indices	Degree	Neighbours indices
0	2	2, 3
1	3	3, 4, 5
2	2	0, 3
3	5	0, 2, 1, 4, 5
4	3	1, 3, 5
5	3	1, 3, 4

Adjacency matrix

- Plain, obvious, less effective
- 2D array, matrix

Node indices	0	1	2	3	4	5
0	0	0	1	1	0	0
1	0	0	0	1	1	1
2	1	0	0	1	0	0
3	1	1	1	0	1	1
4	0	1	0	1	0	1
5	0	1	0	1	1	0

Small graph zoo

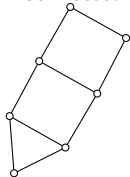
Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

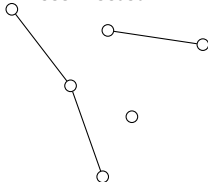
Graphs

Graph representations

Connected

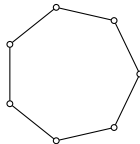


Disconnected



Cycle

- N nodes and N edges



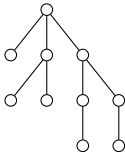
Path

- N nodes and $N - 1$ edges



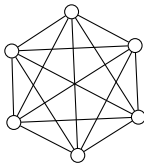
Tree

- Connected, no cycles
- N nodes and $N - 1$ edges



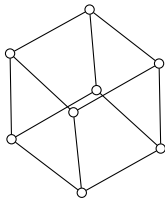
Complete

- Connected
- N nodes and $(N^2 - N)/2$ edges



Regular

- All node degrees are the same



Small graph zoo

Lecture 4:
Graphs,
graph representation,
basic graph processing

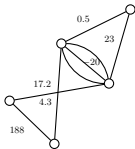
Robert
Pěnička,
Daniel
Průša

Graphs

Graph representation

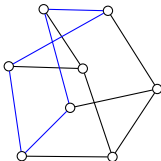
Weighted graph

- Each edge has its cost (length, weight)



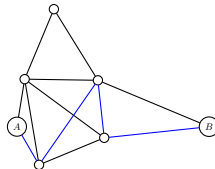
Cycle in a graph

- Path whose first and last node are the same



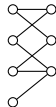
Path between A and B

- Path visits each node at most once



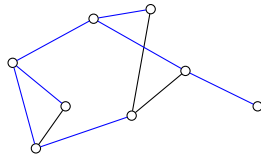
Bipartite graph

- Two-colorable; cycles only of even length; no edges inside partitions



Spanning tree

- Subgraph that is a tree and contains all nodes



Complete bipartite graph

- M and N nodes in partitions;
 $M \times N$ edges



Small graph zoo

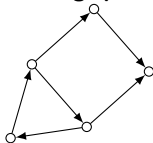
Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

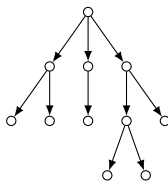
Graphs

Graph representations

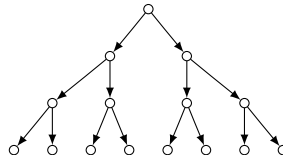
Directed graph



Rooted tree

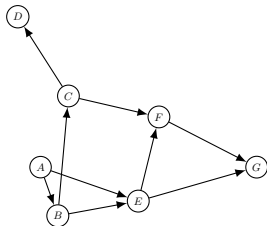


Binary rooted tree



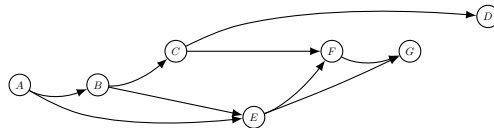
Directed acyclic graph (DAG)

- No directed loops



Topological order

- of the same DAG



A few apparently innocuous problems related to graphs

Easy problem = a complete solution may be taught in bachelor courses.

Hard problem = a complete solution is unknown to this day.

(However, there often exist satisfactory approximate solutions.

Typically, they are quite advanced.)

Clay Mathematics Institute

<https://www.claymath.org/millennium-problems/>

Offers prize **\$1,000,000** for a complete solution of any of those hard questions.

The prize exists since the year 2000.

Nobody has claimed it yet :-(

Graph Connectivity

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

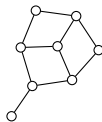
Graph representations

Is there a path between any two nodes?

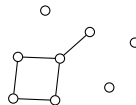
Easy problem

Algorithm: DFS, BFS, Union-Find

Complexity: DFS, BFS $O(|V| + |E|)$, Union-Find $O(|E| \cdot \alpha(|V|))$



Yes,
one connected component.



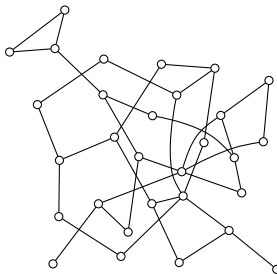
No,
four connected components.

Connectivity

Is there a path between any two nodes?

Easy problem

Is the graph connected?

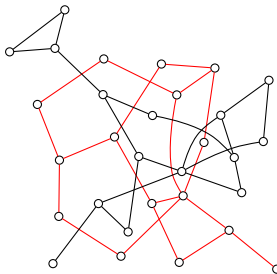


Connectivity

Is there a path between any two nodes?

Easy problem

Is the graph connected? **(No!)**



Independence

Lecture 4:
Graphs,
graph representation,
basic graph processing

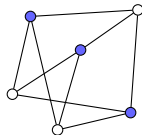
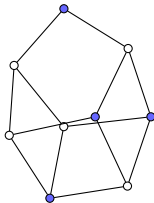
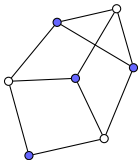
Robert
Pěnička,
Daniel
Průša

Graphs

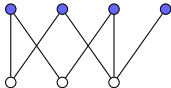
Graph representations

Independence – Maximum size of a set of nodes in which no two nodes are adjacent.

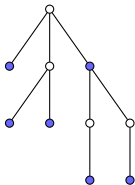
Hard problem in general



Easy problem on graphs with some particular structure

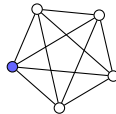


Bipartite graph



Tree (always
bipartite)

Cycle



Complete graph

Independence

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

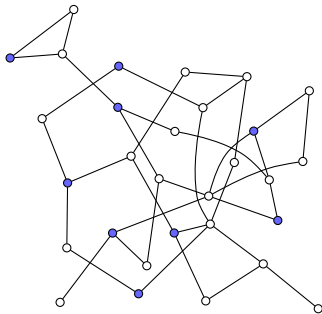
Graphs

Graph representations

Independence – Maximum size of a set of nodes in which no two nodes are adjacent.

Example: How many of them in this graph? more than 9?

Hard problem

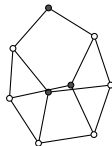
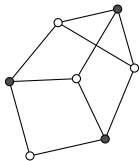


Dominance

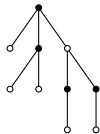
Dominance – Maximum size of such set M of nodes that each node in the graph is either in M or is a neighbour of some node in M .

Example: A fire station must be located either in a village or in the immediately neighbouring village. How many fire stations are enough to serve the region?

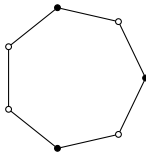
Hard problem



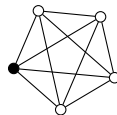
Easy problem on graphs with some particular structure



Tree (apply Dynamic programming)



Cycle



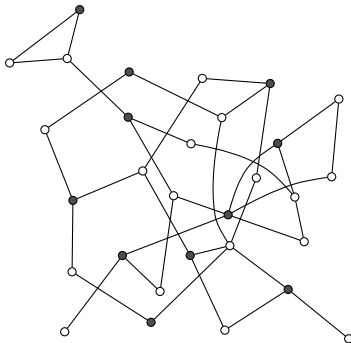
Complete graph

Dominance

Dominance – Maximum size of such set M of nodes that each node in the graph is either in M or is a neighbour of some node in M .

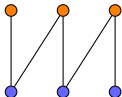
Ex: A fire station must be located either in a village or in the immediately neighbour village. Can there be less than 11 fire stations to serve the region?

Hard problem

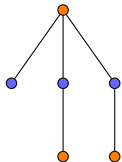


Colorability, chromatic number – Minimum number of colors needed to color each node so that no two neighbours have the same color.

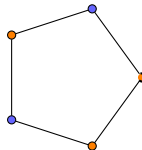
Is 2 colors enough? – Easy problem. Graph must be bipartite.



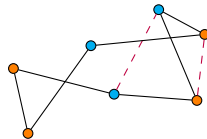
2 colors, bipartite graph



2 colors for any tree



2 colors not enough in odd cycle



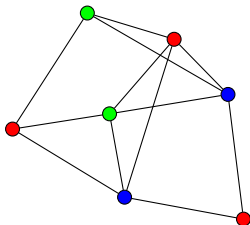
Graph not 2-colorable (odd cycle)

Is graph bipartite? Apply BFS. Mark by 1 all nodes in odd distance from the start and by 0 all nodes in even distance. If two nodes with the same mark are connected by an edge, the graph is not bipartite (two-colorable).

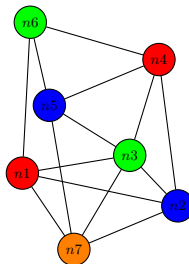
Colorability, chromatic number

Minimum number of colors needed to color each node so that any two neighbours have different color.

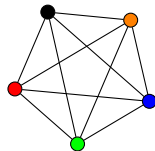
Hard problem – Are 3 colors enough?



3 colors suffice



4 colors. Extra node cannot reuse rgb



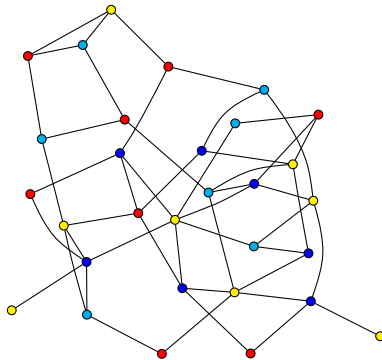
5 colors. Clique of size 5 (hard problem)

Colorability, chromatic number

Minimum number of colors needed to color each node so that any two neighbours have different color.

Hard problem – Are 3 colors enough?

4 colors are sufficient in this graph.
Maybe 3 colors would suffice too?
... ??



Shortest and Longest Paths

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations

Shortest paths

Minimum possible number of edges (nodes) on a path from A to B.

Easy problem

Algorithms: BFS, Dijkstra, Bellman–Ford, Floyd–Warshall, Johnson...

Complexities: Polynomial, mostly less than $O(|V|^3)$.

Longest paths

Typically, each node/edge can be visited at most once.

Hard problem for general graphs

Easy problem for trees and DAGs

Algorithm: Dynamic programming

Complexity: $O(|V| + |E|)$

Minimum spanning tree

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations

Minimum spanning tree

Minimum total cost (weight) of selected edges which connect all nodes in the graph. The selected edges form a tree.

Easy problem

Algorithms:

Prim's $O(|V|^2)$

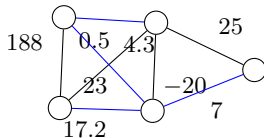
$O(|E| \cdot \log(|V|))$

with matrix representation

with linked list representation and with binary heap

Kruskal's $O(|E| \cdot \log(|V|))$

Borůvka's $O(|E| \cdot \log(|V|))$



Minimum spanning tree

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

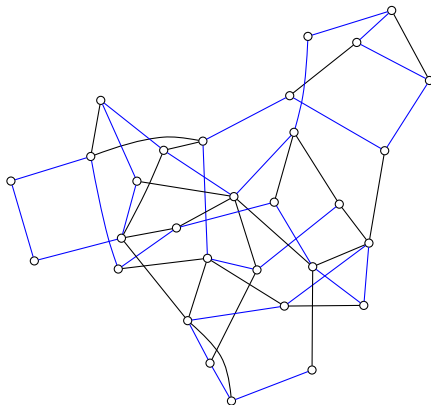
Graphs

Graph representations

Minimum spanning tree

Minimum total cost (weight) of selected edges which connect all nodes in the graph. The selected edges form a tree.

Easy problem



Travelling salesman problem (TSP)

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

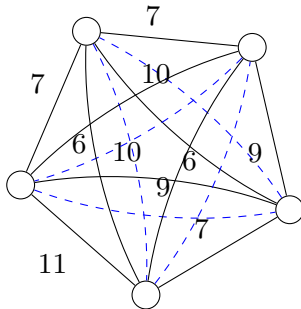
Graphs

Graph representation

Travelling salesman problem (TSP)

Traverse a complete weighted graph, visit each node once and pay the minimum price for the journey
= sum of costs of all visited edges.

Hard problem



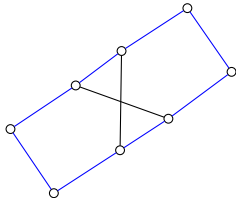
Hamilton path

Is there a path in the graph which contains each node (exactly once)?

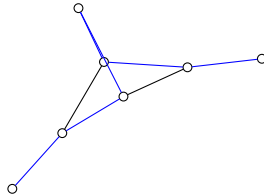
Hamilton cycle

Is there a cycle in the graph which contains each node?

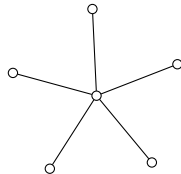
Hard problem



Both Hamilton path and Hamilton cycle exist.



Only Hamilton path exists. There is no Hamilton cycle.



Neither a Hamilton path nor a Hamilton cycle exists.

Euler trail

Euler trail — A trail that visits every edge exactly once (vertices may be revisited).

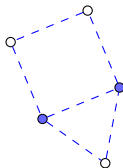
Ex: Can a postman walk through each street in their region exactly once?

Easy problem Graph must be connected **and** it must contain at most **two** nodes of odd degree.

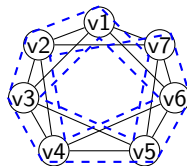
Algorithm: Hierholzer's $O(|E|)$



Euler trail does not exist, there are > 2 nodes with odd degree.



The trail starts and ends in the nodes with odd degree.



The trail is closed; all node degrees are even.

Planar graph

Lecture 4:
Graphs,
graph representation,
basic graph processing

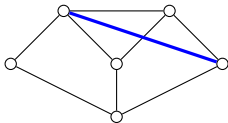
Robert
Pěnička,
Daniel
Průša

Graphs

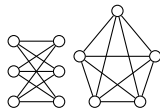
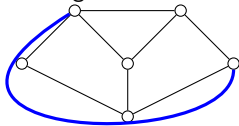
Graph representations

Can the graph be drawn in a plane without crossing its edges?

Easy question (however, little bit more advanced)



The graph is planar, the blue edge can be drawn differently:



Not planar.

Non planar graphs “contain” either

- a **complete graph on 5 nodes**, or
- a **complete bipartite graph on 3 and 3 nodes**.

The planar graphs do not “contain” them.

Algorithm:

Hopcroft and Tarjan, $O(|V|)$

Boyer and Myrvold, $O(|V|)$

Planar graph

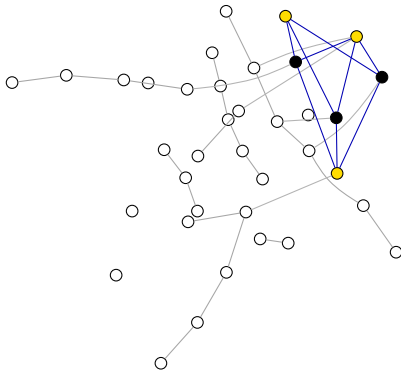
Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations

- Can the graph be drawn in a plane without crossing its edges?



It is impossible here.

- Each black node is connected to each yellow node by a separate path.
- This part is a complete bipartite graph with partitions of size 3 and 3 ($K_{3,3}$).
- Such a graph **cannot be drawn in the plane without edge crossings**.

Clique number

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

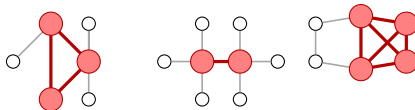
Graphs

Graph representations

The size of the maximal clique, that is, of a subgraph which is complete — that is, of the subgraph where each node is connected to each other node.

Ex. Choose the largest group of your friends in which everybody knows each other.

Hard problem



Clique number of all trees is 2. (Rather obviously.)

Graph isomorphism

Lecture 4:
Graphs,
graph representation,
basic graph processing

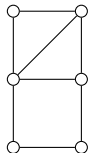
Robert
Pěnička,
Daniel
Průša

Graphs

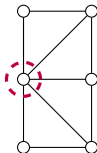
Graph representations

- *Is the structure of two graphs identical?*
- *In other words, can one graph be drawn in such a way that it looks exactly as the other one?*

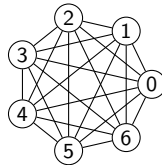
It is not known if this is a hard problem or an easy problem.



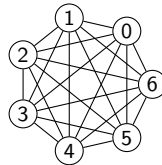
A



B



C



D

A and B are not isomorphic: the right central node in B has degree 5, but there is no such node in A, so their structures differ.

C and D are isomorphic: the nodes with the same labels correspond, and edges connect identical pairs.

Partial recapitulation of the jungle of graph problems and their complexities

Easy problem

- Connectivity?
- Shortest path?
- Min. spanning tree?
- Euler trail?
- Planarity?

"It depends..."

• Colorability?

1,2 colors easy

3 or more colors hard

• Isomorphism?

Trees, ciculants easy

regular graphs, etc... hard

• Longest path?

DAG, tree easy

General graph hard

Hard problem

- Travelling salesman?
- Independence?
- Dominancy?
- Hamiltonicity?
- Clique number?

Many more questions ... ?

Again, "*it depends*". There is no definite cookbook for determining the difficulty of a problem.

Graph most usual representations — Directed weighted graph

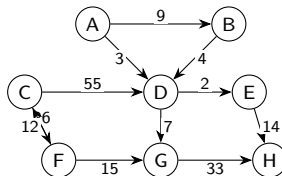
Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations

Directed weighted graph

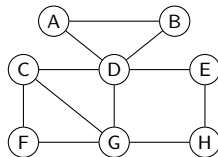


The representation is usually a more or less obvious combination of the methods in the previous cases — **Weight matrix or linked list**.

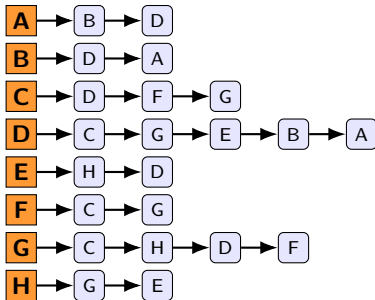
Graph most usual representations — Undirected graph

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša



Linked list representation



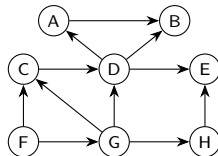
Adjacency matrix

	A	B	C	D	E	F	G	H
A	0	1	0	1	0	0	0	0
B	1	0	0	1	0	0	0	0
C	0	0	0	1	0	1	1	0
D	1	1	1	0	1	0	1	0
E	0	0	0	1	0	0	0	1
F	0	0	1	0	0	0	1	0
G	0	0	1	1	0	1	0	1
H	0	0	0	0	1	0	1	0

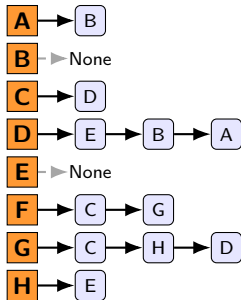
Graph most usual representations — Directed graph

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša



Linked list representation



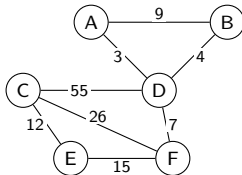
Adjacency matrix

	A	B	C	D	E	F	G	H
A	0	1	0	0	0	0	0	0
B	0	0	0	0	0	0	0	0
C	0	0	0	1	0	0	0	0
D	1	1	0	0	1	0	0	0
E	0	0	0	0	0	0	0	0
F	0	0	1	0	0	0	1	0
G	0	0	1	1	0	0	0	1
H	0	0	0	0	1	0	0	0

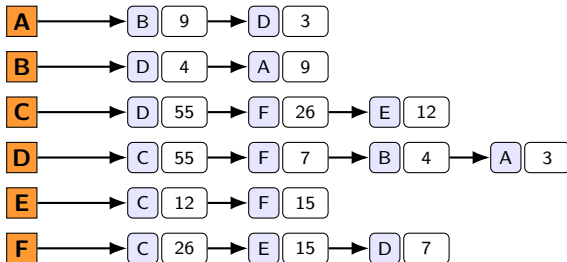
Graph most usual representations — Undirected weighted graph

Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša



Linked list representation

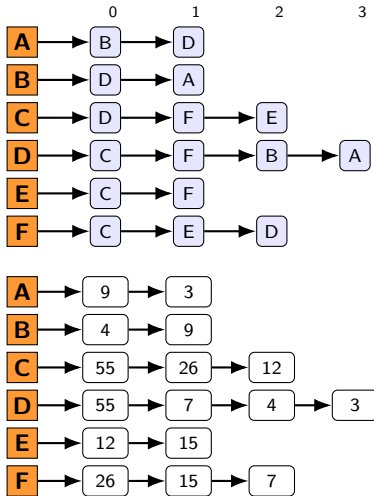


Weight (cost) matrix

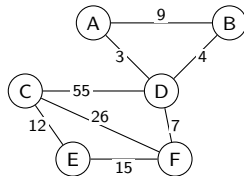
	A	B	C	D	E	F
A	0	9	0	3	0	0
B	9	0	0	4	0	0
C	0	0	0	55	12	26
D	3	4	55	0	0	7
E	0	0	12	0	0	15
F	0	0	26	7	15	0

Graph most usual representations — Undirected weighted graph — alternative rep.

Linked list / array representation



Undirected weighted graph



The weights of edges are at the same index in the second list.

+ **Pro:** Simpler object or even no object at all in arrays.

- **Con:** Keeping lists in sync needs more care and caution in code.

Graph most usual representations — Directed weighted graph

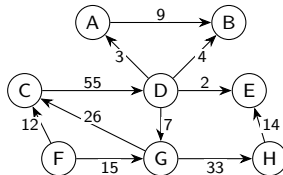
Lecture 4:
Graphs,
graph representation,
basic graph processing

Robert
Pěnička,
Daniel
Průša

Graphs

Graph representations

Directed weighted graph



The representation is usually a more or less obvious combination of the methods in the previous cases
— **Weight matrix or linked list.**