Discrete Mathematics

Graphs

(c) Marcin Sydow

Contents

Introduction

The role of graphs:

- extremely important in computer science and mathematics
- numerous important applications
- modeling the concept of binary relation

Graphs are extensively and intuitively to convey information in visual form.

Here we introduce basic mathematical view on graphs.

Graph (the mathematical definition)

Graph (undirected graph) is an ordered pair of sets: $G = (V, E)$, where:

- V is the *vertex*[1] set
- E is the *edge* set
- each edge $e = \{v, w\}$ in E is an **unordered** pair of vertices from V, called the *ends* of the edge e.

Vertex can be also called **node**.

[1] plural form: *vertices*

Edges and vertices

For an edge $e = \{v, w\} \in E$ we say:

- the edge e *connects* the vertices v i w
- the vertices v and w are *neighbours* or are **adjacent** in the graph G
- the edge e is **incident** to the vertex v (or w).
- a **self-loop** is an edge of the form (v, v).

If V and E are empty G is the *zero graph*, if E is empty it is an *empty* graph

Directed graph (digraph) (mathematical definition)

Directed graph (digraph) is an ordered pair: $G = (V, E)$, where:

- V is the *vertex* set
- E is the *edge* set (or *arc* set)
- each edge $e = (v, w)$ in E is an **ordered** pair of vertices from V, called the *tail* and *head* end of the edge e, respectively.

Example

Simple graphs, multigraphs and hypergraphs

Simple graph: a graph where there are no self-loops (edges or arcs of the form (v, v)).

If there are possible multiple edges or arcs between the same pair of vertices we call it a **multi-graph**.

Notice: in a directed graph (v, w) is a different arc than (w, v) for $v \neq w$.

Picture of a graph

A given graph can be **depicted** on a plane (or other 2-dimensional surface) in multiple ways (example).

A picture is only a **visual form of representation** of a graph.

It is necessary to distinguish between an abstract (mathematical) concept of a graph and its picture (visual representation)

Degree of a vertex

Degree of a vertex v denoted as $deg(v)$ is the number of edges (or arcs) incident with this vertex.

(note: we assume that each self-loop (v, v) contributes 2 to the degree of the vertex v)

If $deg(v) = 0$ we call it an **isolated** vertex.

Example

Degree sum theorem (hand-shake theorem)

The sum of degrees of all vertices in any graph is always even.

(why?)

Degree sum theorem (hand-shake theorem)

The sum of degrees of all vertices in any graph is always even.

(why?)

Proof: each edge contributes 2 to the sum of degrees.

Corollary: sum of degrees is twice the number of edges

Corollary: the number of vertices with odd degree must be even.

Example

Degrees in directed graphs

In directed graphs: **indegree** of a vertex v ($indeg(v)$): number of arcs that v is the head of

outdegree of a vertex v ($outdeg(v)$): number of arcs that v is the tail of

Example

Degree sum theorem for digraphs

The sum of indegrees of all vertices is equal to the sum of outdegrees of all vertices in any directed graph.

Proof: each arc contributes 1 to the indegree sum and 1 to the outdegree sum.

Corollary: sum of indegrees (outdegrees) is equal to the number of arcs in a digraph.

Graph Isomorphism

Two graphs $G_1(V_1, E_1)$, $G_2(V_2, E_2)$ are **isomorphic** $\Leftrightarrow$ there exists a bijection $f : V_1 \rightarrow V_2$ so that:

v, w are connected by an edge (arc) in G_1 $\Leftrightarrow$
$f(v), f(w)$ are connected by an edge (arc) in G_2.

The function f is called **isomorphism** between graphs G_1 and G_2.

Example

Interpretation: graphs are isomorphic if they are "the same" from the point of view of the graph theory (they can have different names of vertices or be differently depicted, etc.).

Subgraph and induced graph

Subgraph of graph $G = (V, E)$ is a graph $H = (V', E')$ so that $V' \subseteq V$ and $E' \subseteq E$ and any edge from E' has both its ends in V'.

Example

A subgraph of G **induced** by a set of vertices $V' \subseteq V$ is a subgraph G' of G whose vertex set is V' whose edges (arcs) are all edges (arcs) of G that have both ends in V'.

Example

Some important graph families

(all graphs below are simple graphs)

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)
- full graph K_n (a simple graph of n vertices and all possible edges (arcs)) (example)

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)
- full graph K_n (a simple graph of n vertices and all possible edges (arcs)) (example)
- bi-partite graph (its set of vertices can be divided into two disjoint sets so that any edges (arcs) are only between the sets) (example)

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)
- full graph K_n (a simple graph of n vertices and all possible edges (arcs)) (example)
- bi-partite graph (its set of vertices can be divided into two disjoint sets so that any edges (arcs) are only between the sets) (example)
- full bi-partite graph $K_{m,n}$ (a bipartite graph that has all possible edges (arcs))

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)
- full graph K_n (a simple graph of n vertices and all possible edges (arcs)) (example)
- bi-partite graph (its set of vertices can be divided into two disjoint sets so that any edges (arcs) are only between the sets) (example)
- full bi-partite graph $K_{m,n}$ (a bipartite graph that has all possible edges (arcs))
- path graph P_n (example)

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)
- full graph K_n (a simple graph of n vertices and all possible edges (arcs)) (example)
- bi-partite graph (its set of vertices can be divided into two disjoint sets so that any edges (arcs) are only between the sets) (example)
- full bi-partite graph $K_{m,n}$ (a bipartite graph that has all possible edges (arcs))
- path graph P_n (example)
- cyclic graph C_n (example)

Some important graph families

(all graphs below are simple graphs)

- empty graph N_n (n vertices, no edges) (example)
- full graph K_n (a simple graph of n vertices and all possible edges (arcs)) (example)
- bi-partite graph (its set of vertices can be divided into two disjoint sets so that any edges (arcs) are only between the sets) (example)
- full bi-partite graph $K_{m,n}$ (a bipartite graph that has all possible edges (arcs))
- path graph P_n (example)
- cyclic graph C_n (example)

Adjacency Matrix

For a graph $G = (V, E)$, having n vertices its **adjacency matrix** is a square matrix A having n rows and columns indexed by the vertices so that $A[i, j] = 1 \Leftrightarrow$ vertices i, j are adjacent, else $A[i, j] = 0$.
(in case of self-loop (i, i), $A[i, i] = 2$)

Example

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i:

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i: degree of i (outdegree for digraphs)

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i: degree of i (outdegree for digraphs)
- sum of numbers in a column i:

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i: degree of i (outdegree for digraphs)
- sum of numbers in a column i: degree of i (indegree for digraphs)

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i: degree of i (outdegree for digraphs)
- sum of numbers in a column i: degree of i (indegree for digraphs)
- for directed graphs A^T reflects the graph

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i: degree of i (outdegree for digraphs)
- sum of numbers in a column i: degree of i (indegree for digraphs)
- for directed graphs A^T reflects the graph with all the arcs "inversed"

Some Simple Observations

Some simple relations concerning properties of a graph and properties of its adjacency matrix:

- for undirected graphs the matrix is symmetric ($A^T = A$)
- for simple graphs the diagonal of A contains only zeros
- sum of numbers in a row i: degree of i (outdegree for digraphs)
- sum of numbers in a column i: degree of i (indegree for digraphs)
- for directed graphs A^T reflects the graph with all the arcs "inversed"

Examples

Incidence matrix

An incidence matrix I of an undirected graph G: the rows correspond to vertices and columns correspond to edges (arcs). $I[v, e] = 1 \Leftrightarrow$ v is incident with e (else I[v,e]=0)

Example

For directed graphs: the only difference is the distinction between v being the head (=1) or the tail (=-1) of e

Example

Graphs vs relations

Each directed graph naturally represents any binary relation $R \in V \times V$. (i.e. E is the set of all pairs of elements from V that are in the relation)

Example

Each undirected graph naturally represents any *symmetric* binary relation

Example

Observations on analogies between relations and graphs

- reflexive relation:

Observations on analogies between relations and graphs

- reflexive relation: self-loop on each vertex
- symmetric relation:

Observations on analogies between relations and graphs

- reflexive relation: self-loop on each vertex
- symmetric relation: undirected graph or always mutual arcs
- transitive relation:

Observations on analogies between relations and graphs

- reflexive relation: self-loop on each vertex
- symmetric relation: undirected graph or always mutual arcs
- transitive relation: for any path there is a “short” arc
- anti-symmetric relation:

Observations on analogies between relations and graphs

- reflexive relation: self-loop on each vertex
- symmetric relation: undirected graph or always mutual arcs
- transitive relation: for any path there is a “short” arc
- anti-symmetric relation: no mutual arcs, always self-loops
- inverse of the relation:

Observations on analogies between relations and graphs

- reflexive relation: self-loop on each vertex
- symmetric relation: undirected graph or always mutual arcs
- transitive relation: for any path there is a "short" arc
- anti-symmetric relation: no mutual arcs, always self-loops
- inverse of the relation: each arc is inversed

Path

Path: an alternating sequence of vertices and edges $(v_0, e_0, v_1, e_q, \ldots, v_k, e_k, \ldots, v_l)$ so that each edge e_k is incident with vertices v_k, v_{k+1}. We call it a **path from** v_0 **to** v_l.

(sometimes it is convenient to define path just as a subsequence of vertices or edges of the above sequence)

Example

Directed path in a directed graph is defined analogously (the arcs must be directed from v_k to v_{k+1}

Paths cont.

simple path: no repeated edges (arcs)

elementary path: no repeated vertices

Examples

length of a path: number of its edges (arcs)

(assume: 0-length path is a single vertex)

Example

Distance in graph

Distance between two vertices is the length of a shortest path between them.

The distance function in graphs $d : V \times V \rightarrow N$ has the following properties:

- $d(u, v) = 0 \Leftrightarrow u == v$
- (only in undirected graphs) it is a symmetric function, i.e. $\forall u, v \in V$ d(u,v) = d(v,u)
- triangle inequality: $\forall u, v, w \in V$ it holds that $d(u, v) + d(v, w) \geq d(u, w)$

Cycle

cycle: a path of length at least 3 (2 for directed graphs) where the beginning vertex equals the ending vertex $v_0 == v_l$ (also called a closed path)

Example

analogously: **directed cycle**, **simple cycle**, **elementary cycle** (except the starting and ending vertices there are no repeats)

Examples

Connectedness

A graph is **connected** $\Leftrightarrow$ for any two its vertices v,w there exists a path from v to w

Example

Connected component of a graph

Connected component of a graph is its maximal subgraph that is connected.

Example (why "maximal")?

Strongly connected graph

(only for directed graphs)

A directed graph is **stronlgy connected** $\Leftrightarrow$ for any pair of its vertices v,w there exists a directed path from v to w.

Example

A directed graph is **weakly connected** $\Leftrightarrow$ for any pair of its vertices v,w there exists *undirected* path from v,w (i.e. the directions of arcs can be ignored)

note: strong connectedness implies weak connectedness (but not the opposite)

Example

Strongly and weakly connected components

Strongly connected component: a maximal subgraph that is strongly connected

Weakly connected component: a maximal subgraph that is weakly connected

Examples

Drzewa

Tree is a graph that is connected and does not contain cycles (acyclic).

Example

Forest is a graph that does not contain cycles (but does not have to be connected)

Example

A **leaf** of a tree is a vertex that has degree 1.

Other vertices (nodes) are called **internal nodes** of a tree.

Example

Equivalent definitions of a tree

The following conditions are equivalent:

Equivalent definitions of a tree

The following conditions are equivalent:

- T is a tree of n vertices

Equivalent definitions of a tree

The following conditions are equivalent:

- T is a tree of n vertices
- T has exactly n-1 edges (arcs) and is acyclic

Equivalent definitions of a tree

The following conditions are equivalent:

- T is a tree of n vertices
- T has exactly n-1 edges (arcs) and is acyclic
- T is connected and has exactly n-1 edges (arcs)

Equivalent definitions of a tree

The following conditions are equivalent:

- T is a tree of n vertices
- T has exactly n-1 edges (arcs) and is acyclic
- T is connected and has exactly n-1 edges (arcs)
- T is connected and removing any edge (arc) makes it not connected

Equivalent definitions of a tree

The following conditions are equivalent:

- T is a tree of n vertices
- T has exactly n-1 edges (arcs) and is acyclic
- T is connected and has exactly n-1 edges (arcs)
- T is connected and removing any edge (arc) makes it not connected
- any two vertices in T are connected by exactly one elementary path

Equivalent definitions of a tree

The following conditions are equivalent:

- T is a tree of n vertices
- T has exactly n-1 edges (arcs) and is acyclic
- T is connected and has exactly n-1 edges (arcs)
- T is connected and removing any edge (arc) makes it not connected
- any two vertices in T are connected by exactly one elementary path
- T is acyclic and adding any edge makes exactly one cycle

Rooted tree

A **rooted tree** is a tree with exactly one distinguished node called its **root**.

Example

Distinguishing the root introduces a **natural hierarchy** among the nodes of the tree: the lower the depth the higher the node in the hierarchy.

Picture of a rooted tree: root is at the top, all nodes of the same depth are on the same level, the higher the depth, the lower the level on the picture.

Example

Terminology of rooted trees

A **depth** of a vertex v of a rooted tree, denoted as $depth(v)$ is its distance from the root.

Height of a rooted tree: maximum depth of any its node

ancestor of a vertex v is any vertex w that lies on any path from the root to v, v is then called a **descendant** of w (the root does not have ancestors and the leaves do not have descendants)

a ancestor w of a neighbour (adjacent) vertex v is called the **parent** of v, in this case v is called the **child** of w.

if vertices u, v have a common parent we call them **siblings**

Examples

Binary tree

Binary tree is a rooted tree with the following properties:

- each node has maximally 2 children
- for each child it is specified whether it is left or right child of its parent (max. 1 left child and 1 right child)

Example

Summary

- Mathematical definition of Graph and Digraph
- Degree of a vertex
- Graph isomorphism
- Adjacency and Incidence Matrices
- Graphs vs Relations
- Path and Cycle
- Connectedness
- Weakly and strongly connected components
- Tree, Rooted tree, Binary tree

Example tasks/questions/problems

- give the mathematical definitions and basic properties of the discussed concepts and their basic properties (in particular: graph, digraph, degree, isomorphism, adjacency/incidence matrix, path and cycle, connectedness and connected components, trees (including rooted and binary trees)
- make picture of the specified graph of one of the discussed families (full, bi-partite, etc.)
- given a picture of a graph provide its mathematical form (pair of sets) and adjacency/incidence matrix and vice versa
- check whether the given graphs are isomorphic and prove your answer
- find connected components of a given graph (or weakly/strongly connected components for a digraph)
- specify the height, depth, number of leaves, etc. of a given rooted tree

Thank you for your attention.