GRAPHS

Definition

A graph is a pair $G = (V, E)$ where V is a nonempty finite set and E is a set of two–element subsets of V.

$G = (V, E)$
$V = \{v_1, v_2, v_3, v_4\}$, $E = \{\{v_1, v_3\}, \{v_2, v_4\}, \{v_1, v_4\}, \{v_2, v_3\}\}$

The elements of V are called the vertices of the graph, and the elements of E are called the edges of the graph.
Let G be a graph. If we neglect to give a name to the vertex set and edge set of G, we can simply write $V(G)$ and $E(G)$ for the vertex and edge sets, respectively.

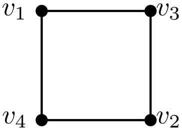
GRAPHS

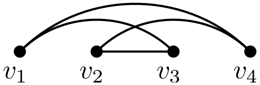
How to draw pictures of graphs? These pictures make graphs much easier to understand.

A drawing of the graph $G = (V, E)$ is a mapping that assing a point in the plane for each vertex and for each edge a continuous curve between its two endpoints.

A drawing of the graph is not the same thing as the graph itself.

The following two drawings both depict the same graph.

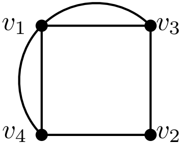




Definition

A *multigraph* $G = (V, E)$ *consists of a set of vertices* V, *a set of edges* E, *and a function* f *from* E *to* $\{\{u, v\} : u, v \in V, u \neq v\}$. *The edges* e_1 *and* e_2 *are called* *multiple* *or* *parallel edges* *if* $f(e_1) = f(v_2)$.

$G = (V, E)$
$V = \{v_1, v_2, v_3, v_4\}$, $E = \{\{v_1, v_3\}, \{v_2, v_4\}, \{v_1, v_4\}, \{v_2, v_3\}, \{v_1, v_4\}\{v_1, v_3\}\}$



Definition

Two vertices u and v in a graph $G = (V, E)$ are called adjacent in G if $\{u, v\}$ is an edge of G.
If $e = \{u, v\}$, the edge e is called incident with the vertices u and v.

If $\{u, v\}$ is an edge of G, we call u and v the endpoints of the edge.

Definition

Let $G = (V, E)$ be a graph and let $v \in V$. The degree of v is the number of edges with which v is incident. The degree of v is denoted $deg(v)$ or $\delta(v)$.

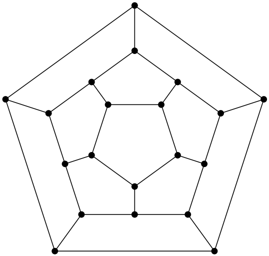
Theorem

Let $G = (V, E)$. The sum of the degrees of the vertices in G is twice the number of edges, that is,

$$\sum_{v \in V} deg(v) = 2 \cdot |E|$$

Definition

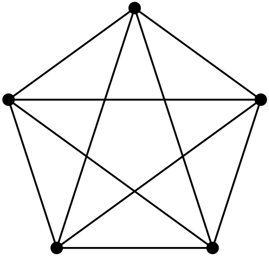
If all vertices in G have the same degree, we call G regular. If a graph is regular and all vertices have degree r, we also call the graph r–regular.



GRAPHS

Definition

Let G be a graph. If all pairs of distinct vertices are adjacent in G, we call G complete. A complete graph on n vertices is denoted K_n.

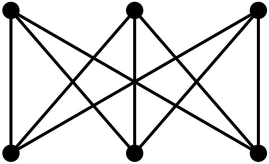


The opposite extreme is a graph with no edges. We call such graphs edgeless.

Definition

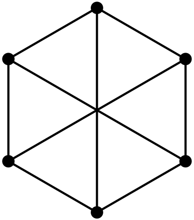
Let $m, n \in \mathbb{N}$. *The* *complete bipartite graph*, $K_{m,n}$, *is a graph whose vertices can be partitioned* $V = V_1 \cup V_2$ *such that*

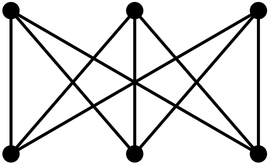
- $|V_1| = m$ *and* $|V_2| = n$
- *for all* $u \in V_1$ *and for all* $v \in V_2$, $\{u, v\}$ *is an edge.*



Definition

Let $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$ be graphs. We say that G_1 is isomorphic to G_2 provided there is a bijection $f : V_1 \to V_2$ such that for all $u, v \in V_1$ we have $\{u, v\} \in E_1$ if and only $\{f(u), f(v)\} \in E_2$. The function f is called an isomorphism of G_1 to G_2.





GRAPHS

Definition

Let $G = (V, E)$ and $G_1 = (V_1, E_1)$ be graphs. We call G_1 a subgraph of G provided $V_1 \subseteq V$ and $E_1 \subseteq E$.

Definition

Let $G = (V, E)$ be a graph. We call $G_1 = (V_1, E_1)$ a spanning subgraph of G provided $V_1 = V$ and $E_1 \subseteq E$.

Definition

Let G be a graph. The complement of G is the graph denoted $\overline{G}$ defined by

$$V(\overline{G}) = V(G)$$

$$E(\overline{G}) = \{\{u, v\} : u, v \in V(G), u \neq v, \{u, v\} \notin E(G)\}$$

.

Definition

Let $G = (V, E)$ be a graph. A walk of length n ($n \in \mathbb{N}$) in G is a sequence of vertices $v_0, v_1, \ldots v_n$ of the graph such that $\{v_0, v_1\}, \{v_1, v_2\}, \ldots \{v_{n-1}, v_n\}$ are edges, where $v_0 = u$ and $v_n = v$.
A path of length n in a graph is a walk in which no vertex is repeated.
A path on n edges is denoted P_n.
A cycle is a path of length at least three in which the first and the last vertex are the same, but no other vertices are repeated.
A cycle on n edges is denoted C_n.
A length of a walk (path, cycle) is the number of edges on that walk (path, cycle).

(u, v)–path

Definition

A graph $G = (V, E)$ is called connected provided for all $u, v \in V$ there is (u, v)–path.

A connected graph consists of one "piece", while a graph that is not connected consists of two or more "pieces". These "pieces"we called components of the graph.

Definition

Let $G = (V, E)$ be a graph and let $u, v \in V$. The distance from u to v in G is the length of the shortest (u, v)–path. In case there is no such a path, we may either say that the distance is undefined or distance is ∞. The distance from u to v is denoted $d(u, v)$.

Definition

Let $s_1, s_2, s_3, \ldots, s_n$ be nonnegative integer numbers. Sequence $s_1, s_2, s_3, \ldots, s_n$ is called graphical if there is a graph with n vertices whose degrees are $s_1, s_2, s_3, \ldots, s_n$.

Theorem

Havel's Theorem

Let $s_1,\ s_2,\ s_3,\ \ldots,\ s_n$ $n \geq 2,\ 1 \leq s_1 \leq n-1$ be nonnegative integer numbers so that $s_1 \geq s_2 \geq s_3 \geq \cdots \geq s_n$. This sequence is graphical if and only if the sequence $s_2 - 1,\ s_3 - 1,\ \ldots,\ s_{s_1+1} - 1,\ s_{s_1+2},\ \ldots,\ s_n$ is graphical.

Note that in the sequence $s_1, s_2, s_3, \ldots, s_n$ we delete number s_1 and the following s_1 members will be reduced by 1.

Example

Decide whether a sequence 1, 1, 1, 2, 3, 3, 3, 4, 4, 4, 4 is graphical. If it is graphical sequence, sketch a drawing of a graph.

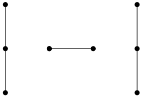
Solution:

$\boxed{4}$, $\underline{4, 4, 4, 3}$, 3, 3, 2, 1, 1, 1	$n = 11$, $s_1 = 4$,
3, 3, 3, 2, 3, 3, 2, 1, 1, 1	we arrange the members
$\boxed{3}$, $\underline{3, 3, 3}$, 3, 2, 2, 1, 1, 1	$n = 10$, $s_1 = 3$,
2, 2, 2, 3, 2, 2, 1, 1, 1	we arrange the members
$\boxed{3}$, $\underline{2, 2, 2}$, 2, 2, 1, 1, 1	$n = 9$, $s_1 = 3$,
1, 1, 1, 2, 2, 1, 1, 1	$n = 8$

GRAPHS

The last sequence 1, 1, 1, 2, 2, 1, 1, 1 is graphical, because there is a graph, denoted by G_1, with 8 vertices whose degrees are 1, 1, 1, 2, 2, 1, 1, 1.

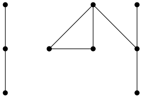
G_1

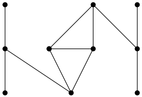


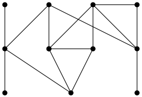
So, the first sequence 4, 4, 4, 4, 3, 3, 3, 2, 1, 1, 1 is graphical too.

Now we sketch a drawing of the graph with 11 vertices whose degrees are 1, 1, 1, 2, 3, 3, 3, 4, 4, 4, 4.

G_2: 2, 2, 2, 3, 2, 2, 1, 1, 1 G_3: 3, 3, 3, 2, 3, 3, 2, 1, 1, 1 G_4: 4, 4, 4, 4, 3, 3, 3, 2, 1, 1, 1







Definition

Suppose that $G = (V, E)$ is a graph where $V = \{v_1, v_2, \ldots, v_n\}$. The adjacency matrix B of G is the $n \times n$ zero–one matrix with 1 as its (i, j)th entry when v_i and v_j are adjacent, and 0 as its (i, j)th entry when they are not adjacent. In another words, if its adjacency matrix is $B = (b_{ij})$, then

$$b_{ij} = \begin{cases} 1 & \text{if } \{v_i, v_j\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

The adjacency matrix of a graph is symmetric, that $b_{ij} = b_{ji}$, since both of these entries are 1 when v_i and v_j are adjacent, and both are 0 otherwise.

GRAPHS AND MATRICES

Definition

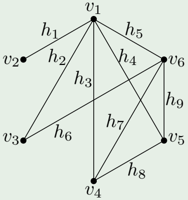
Let $G = (V, E)$ be a graph. Suppose that $v_1, v_2, \ldots, v_n$ are vertices and $e_1, e_2, \ldots, e_m$ are the edges of G. Then the incidence matrix with respect to this ordering of V and E is the $n \times m$ matrix $A = (a_{ij})$ where

$$a_{ij} = \begin{cases} 1 & \text{if } e_j = \{v_i, v_k\} \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Note that in the incidence matrix A of a graph each column has two 1's and that the sum of a row gives the degree of the vertex identified with that row.

Example

Write the adjacency matrix B and the incidence matrix A for the graph G



Solution: Vertices and edges are denoted.

Incidence matrix:

$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{array} \begin{array}{c} \begin{array}{ccccccccc} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 & h_9 \end{array} \\ \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} \end{array}.$$

Adjacency matrix:

$$B = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \\ v_6 \end{array} \begin{array}{c} \begin{array}{cccccc} v_1 & v_2 & v_3 & v_4 & v_5 & v_6 \end{array} \\ \begin{pmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \end{pmatrix} \end{array}.$$

Let B be adjacency matrix of a connected graph $G = (V, H)$, $|V| = n$. Let $B^{(1)}$ be the matrix obtained from B by replacing 0's by 1's on the main diagonal. Thus, $B^{(1)} = B + E$, where E is the identity matrix of the same size. For $k \geq 1$, define zero–one matrix

$$B^{(k)} = B^{(k-1)} \cdot B^{(1)}$$

such that its (i, j)th entry

$$b_{ij}^{(k)} = \sum_{r=1}^{n} b_{ir}^{(k-1)} \cdot b_{rj}^{(1)}$$

is 1 if and only if there is at least one r, $r \in \{1, 2, \ldots n\}$, for which both $b_{ir}^{(k-1)} = 1$ and $b_{rj}^{(1)} = 1$.

Theorem

Let B be the adjacency matrix of a connected graph $G = (V, E)$, $|V| = n$. Then for any k, $k = 1, 2, \ldots, n$, the (i, j)th entry of the matrix $B^{(k)}$ equals 1 if and only if $d(v_i, v_j) \leq k$.

Note that the (i, j)th entry of the matrix $B^{(k)}$ equals 0 if and only if $d(v_i, v_j) > k$.

Theorem

Let $G = (V, E)$, $|V| = n$ be a graph. The graph G is connected if and only if all entries of matrix $B^{(n-1)}$ are equal to 1.

Example

Let

$$B = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

be adjacency matrix of the graph. Without drawing a graph, determine pairs of vertices whose distance is

a) greater than 2,

b) less than or equal to 3,

c) less than 2,

d) equal to 3.

Is a given graph connected?

Solution: $|V| = 6$

$$B^{(1)} = B + E = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

$$B^{(2)} = B^{(1)} \cdot B^{(1)} = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}.$$

$$B^{(3)} = B^{(2)} \cdot B^{(1)} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

$$B^{(4)} = B^{(3)} \cdot B^{(1)} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix}.$$

$$B^{(5)} = B^{(4)} \cdot B^{(1)} = B^{(4)}$$

a) Distance $d(v_i, v_j) > 2$ if and only if $b_{ij}^{(2)} = 0$.
It applies to distances: $d(v_1, v_4)$, $d(v_2, v_4)$, $d(v_2, v_6)$, $d(v_4, v_5)$, $d(v_5, v_6)$.

b) Distance $d(v_i, v_j) \leq 3$ if and only if $b_{ij}^{(3)} = 1$.
It applies to all distances except $d(v_2, v_4)$ and $d(v_4, v_5)$.

c) Distance $d(v_i, v_j)2$ (t. j. $d(v_i, v_j) \leq 2$) if and only if $b_{ij}^{(1)} = 1$.
It applies to distances: $d(v_1, v_1)$, $d(v_1, v_2)$, $d(v_1, v_3)$, $d(v_1, v_5)$, $d(v_2, v_2)$, $d(v_3, v_3)$, $d(v_3, v_6)$, $d(v_4, v_4)$, $d(v_4, v_6)$, $d(v_5, v_5)$, $d(v_6, v_6)$.

d) Distance $d(v_i, v_j) = 3$ (t. j. $d(v_i, v_j) \leq 3$ and $d(v_i, v_j) > 2$) if and only if $b_{ij}^{(2)} = 0$ and $b_{ij}^{(3)} = 1$.
$d(v_1, v_4)$, $d(v_2, v_6)$, $d(v_5, v_6)$.

As matrix $B^{(5)}$ contains only 1's, then the given graph is connected.

Definition

A tree is a connected graph with no cycles.

Definition

A leaf of a graph is a vertex of degree 1.

Theorem

Every tree with at least two vertices has a leaf.

Theorem

Let T be a tree. For any two vertices u and v in $V(T)$, there is a unique (u, v)–path. Conversely, if G is a graph with the property that for any two vertices u, v there is exactly one (u, v)–path, then G must be a tree.

TREES

Theorem

Let T be a tree with $n \geq 1$ vertices. Then T has $n - 1$ edges.

Theorem

Let T be a graph with $n \geq 1$ vertices. The following are equivalent.

(a) *T is a tree.*

(b) *T is connected without cycles.*

(c) *T is connected and has $n - 1$ edges.*

(d) *T has no cycles and has $n - 1$ edges.*

SPANNING TREES

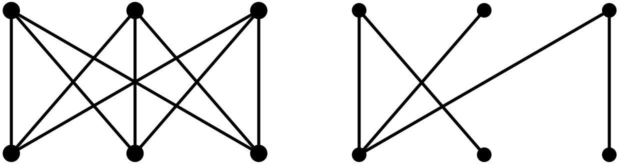
Definition

Let G be a graph. A spanning tree of G is a spanning subgraph of G that is a tree.

Theorem

A graph has a spanning tree if and only if it is connected.

Graph $K_{3,3}$ and its spanning tree.



Theorem

Let B be an adjacency matrix for a graph $G = (V, E)$, $|V| = n$. Assume a $n \times n$ matrix $D = (d_{ij})$ where

$$d_{ij} = \begin{cases} 0 & \text{if } i \neq j, \\ \delta(v_i) & \text{if } i = j. \end{cases}$$

The number of spanning trees of G, denoted $p(T)$, is determine by formula

$$p(T) = \det(D - B)_i,$$

where $(D - B)_i$ is a matrix $D - B$ without i-th row and i-th column.

COLOURING

Definition

Let G be a graph and let k be a positive integer. A k–colouring of G is a function

$$f : V(G) \to \{1, 2, \ldots, k\}.$$

We call this colouring proper provided

$$\forall \{x, y\} \in E(G) : f(x) \neq f(y).$$

If a graph has a proper k–colouring, we call it k–colourable.

To each vertex v the function f associates a value $f(v)$. The value $f(v)$ is a colour of v. The palette of colours we use in the set $\{1, 2, \ldots, k\}$.
The condition $\forall \{x, y\} \in E(G) : f(x) \neq f(y)$ means that whenever vertices x and y are adjacent, then these vertices must get different colours. In a proper colouring, adjacent vertices are not assigned the same colour.
The goal in graph colouring is to use as few colours as possible.

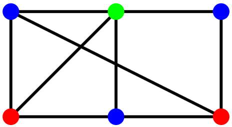
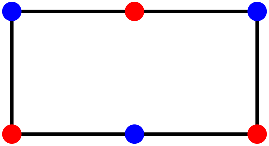
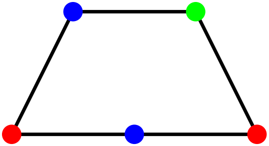
COLOURING

Definition

Let G be a graph. The smallest possitive integer k for which G is k–colourable is called the chromatic number of G.
The chromatic number of G is denoted $\chi(G)$.

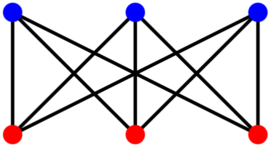
Theorem

Let G be a graph with maximum degree Δ. Then $\chi(G) \leq \Delta + 1$. The chromatic number of G is denoted .

Definition

A graph G is called ***bipartite*** *provided it is 2-colorable.*



Theorem

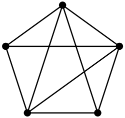
A graph is bipartite if and only if it does not contain an odd cycle.

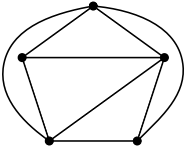
PLANAR GRAPHS

Definition

A planar graph is a graph that has a drawing in the plane in which two edges do not intersect (except at an endpoint if they both are incident with the same vertex).
A graph that is not planar is called nonplanar.

A planar graph has crossing-free embedding in the plane.
A face is a portion of the plane cut off by the drawing. Imagine the graph drawn on a physical piece of paper. If we cut along the curves representing the edges of G, the paper falls apart into various pieces. Each of these pieces is called a face (or region) of the embedding.





The drawing of the graph in the figure has six faces. There are five bounded faces (faces with only finite area) and one unbounded face that surrounds the graph.
The degree of the face is called the number of edges that are on the boundary of

PLANAR GRAPHS

Theorem

Euler's Theorem

Let G be a connected planar graph with n vertices and m edges. Choose a crossing-free embedding for G, and let f be the number of faces in the drawing. Then

$$n - m + f = 2.$$

Corollary

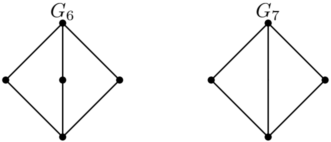
1. *Let G be a planar graph with at least 2 vertices. Then* $|E(G)| \leq 3 \cdot |V(G)| - 6$.
2. *Let G be a planar graph with at least 2 vertices and G does not contain K_3 as a subgraph. Then* $|E(G)| \leq 2 \cdot |V(G)| - 4$.
3. *Let G be a planar graph. Then G contains a vertex with degree at most five.*
4. *The graph $K_{3,3}$ is nonplanar.*
5. *The graph K_5 is nonplanar.*

PLANAR GRAPHS

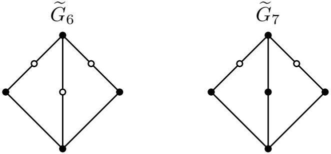
If in the graph we remove an edge $\{u, v\}$ and add a new vertex w together with edges $\{u, w\}$ and $\{w, v\}$, we called such an operation elementary subdivision. A subdivision of G is formed from G by replacing edges with paths (we apply the elementary subdivision operation several times). Graph and its subdivision are called homeomorphic.

If a graph is planar, so are its subdivisions. And the converse of this statement is also true: If a graph is nonplanar, then all of its subdivisions are also nonplanar.

Graphs G_6 and G_7



are homeomorphic because

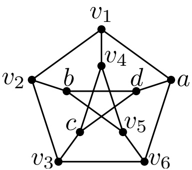


PLANAR GRAPHS

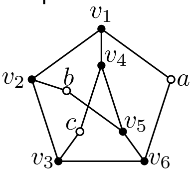
Theorem

Kuratowski's Theorem

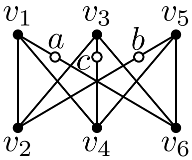
A graph is planar if and only if it does not contain a subdivision of $K_{3,3}$ or K_5 as a subgraph.

Graph 

is nonplanar because contains subgraph



which is subdivision of $K_{3,3}$



PLANAR GRAPHS

Kuratowski's Theorem is a marvelous characterization of planarity.

If a graph is planar, I can convince you of this fact by presenting you with a crossing-free drawing.

On the other hand, if a graph is nonplanar, I can convince you of this fact by finging a subdivision of $K_{3,3}$ or K_5 as a subgraph of my graph.

DIGRAPHS

Definition

A *digraph (directed graph)* $\overrightarrow{G} = (V, E)$ *consists of set of vertices* $V = \{v_1, v_2, \ldots, v_n\}$ *and set of edges* E*, that is subset of the set* $V \times V - \{(v_1, v_1), \ldots, (v_n, v_n)\}$*. If* (v_i, v_j) *is an edge, then the vertex* v_i *is called the* *initial vertex* *and the vertex* v_j *is called the* *terminal vertex*.

Definition

Let $\overrightarrow{G} = (V, E)$ *be a digraph and let* $v \in V$*. The* *out-degree* *of* v*, denoted* $\delta^+(v)$*, is the number of edges for which vetrex* v *is the initial vertex. The* *in-degree* *of* v*, denoted* $\delta^-(v)$*, is the number of edges for which vetrex* v *is the terminal vertex.*
The vertex v *is called a* *source* *if* $\delta^+(v) > 0$ *and* $\delta^-(v) = 0$*.*
The vertex v *is called a* *sink* *if* $\delta^+(v) = 0$ *and* $\delta^-(v) > 0$*.*

DIGRAPHS

Definition

A *directed walk* *(or more simply, a walk) in a directed graph* $\overrightarrow{G}$ *is a sequence of vertices* $v_0, v_1, \ldots, v_k$ *and edges*

$$(v_0, v_1), (v_1, v_2) \ldots, (v_{k-1}, v_k)$$

such that (v_{i-1}, v_i) *is an edge of* $\overrightarrow{G}$ *for all* i *where* $1 \leq i < k$*.*
A *directed path* *(or path) in a directed graph is a walk where the vertices in the walk are all different.*
A *directed cycle* *(or cycle) in a directed graph is a closed walk where all the vertices* v_i *are different for* $1 \leq i < k$*.*

The notion of being connected is a little more complicated for a directed graph than it is for an undirected graph. For example, should we consider the graph in Figure to be connected? There is a path from node a to every other node so on that basis, we might answer „Yes." But there is no path from nodes b, c, or d to node a, and so on that basis, we might answer „No." For this reason, graph theorists have come up with the notion of strong connectivity for directed graphs.

Definition

A directed graph $\overrightarrow{G} = (V, E)$ *is said to be strongly connected if for every pair of vertices* $u; v \in V;$ *, there is a directed path from* u *to* v *(and vice-versa) in* $\overrightarrow{G}$.

For example, the graph in Figure is not strongly connected since there is no directed path from node b to node a. But if node a is removed, the resulting graph would be strongly connected.

Definition

A directed graph $\overrightarrow{G} = (V, E)$ *is said to be* *weakly connected* *(or, more simply, connected) if the corresponding undirected graph/multigraph (where directed edges* (u, v) *and/or* (v, u) *are replaced with a single undirected edge* $\{u, v\}$ *is connected. For example, the graph in Figure is weakly connected.*

Definition

Let $\overrightarrow{G}_1 = (V_1, E_1)$ *and* $\overrightarrow{G}_2 = (V_2, E_2)$ *be digraphs. We say that* $\overrightarrow{G}_1$ *is isomorphic to* $\overrightarrow{G}_2$ *provided there is a bijection* $f : V_1 \to V_2$ *such that for all* $u, v \in V_1$ *we have* $(u, v) \in E_1$ *if and only* $(f(u), f(v)) \in E_2$*. The function* f *is called an isomorphism of* $\overrightarrow{G}_1$ *to* $\overrightarrow{G}_2$*.*

Definition

A digraph is called acyclic if it does not contain a directed cycle.

Theorem

Let $\overrightarrow{G} = (V, E)$ be acyclic digraph. Then there is a vertex $v \in V$ which is a source.

Theorem

A digraph $\overrightarrow{G} = (V, E)$ is acyclic if and only if it is possible denoted the vertices by numbers $1, 2, \ldots, |V|$ such that for every edge (i, j) we have $i < j$.

DIGRAPHS AND MATRICES

Definition

Suppose that $\overrightarrow{G} = (V, E)$ *is a digraph where* $V = \{v_1, v_2, \ldots, v_n\}$. *The adjacency matrix* B *of* $\overrightarrow{G}$ *is the* $n \times n$ *zero–one matrix with 1 as its* (i, j)*th entry when* $(v_i, v_j) \in E$, *and 0 as its* (i, j)*th entry when* $(v_i, v_j) \notin E$. *In another words, if its adjacency matrix is* $B = (b_{ij})$, *then*

$$b_{ij} = \begin{cases} 1 & \text{if } (v_i, v_j) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

The adjacency matrix of a digraph, in generally, is not symmetric.

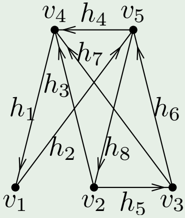
Definition

Let $\overrightarrow{G} = (V, E)$ be a graph. Suppose that $v_1, v_2, \ldots, v_n$ are vertices and $e_1, e_2, \ldots, e_m$ are the edges of $\overrightarrow{G}$. Then the incidence matrix with respect to this ordering of V and E is the $n \times m$ matrix $A = (a_{ij})$ where

$$a_{ij} = \begin{cases} 1 & \text{if } e_j = (v_i, v_k) \in E, \\ -1 & \text{if } e_j = (v_k, v_i) \in E, \\ 0 & \text{otherwise.} \end{cases}$$

Example

Write the adjacency matrix B and the incidence matrix A for the digraph $\overrightarrow{G}$



Solution: Vertices and edges are denoted.
Incidence matrix:

$$A = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{cccccccc} h_1 & h_2 & h_3 & h_4 & h_5 & h_6 & h_7 & h_8 \end{array} \\ \begin{pmatrix} -1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & -1 & 1 & 1 & 0 \\ 1 & 0 & -1 & -1 & 0 & 0 & -1 & 0 \\ 0 & -1 & 0 & 1 & 0 & -1 & 0 & 1 \end{pmatrix} \end{array}.$$

Adjacency matrix:

$$B = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{array} \begin{array}{c} \begin{array}{ccccc} v_1 & v_2 & v_3 & v_4 & v_5 \end{array} \\ \begin{pmatrix} 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 \end{pmatrix} \end{array}.$$

DIRECTED TREES

Definition

Directed tree $\overrightarrow{T}$ *is a digraph which is tree which after cancelling the direction of the edges is a tree.*

Definition

Let $\overrightarrow{G}$ *be a digraph which was created by the direction of the edges of the graph* G*. Let* K *be a spanning tree of the graph* G*. Then the digraph* $\overrightarrow{K}$ *which was created by the direction of the edges of the graph* K*, is called directed spanning tree.*

Theorem

Let B be an adjacency matrix for a digraph $\overrightarrow{G} = (V, E)$, $|V| = n$. Assume a $n \times n$ matrix $D = (d_{ij})$ where

$$d_{ij} = \begin{cases} 0 & \text{if } i \neq j, \\ \delta^+(v_i) + \delta^-(v_i) & \text{if } i = j. \end{cases}$$

The number of directed spanning trees of $\overrightarrow{G}$, denoted $p(\overrightarrow{T})$, is determine by formula

$$p(\overrightarrow{T}) = \det(D - B - B^T)_i,$$

where $(D - B - B^T)_i$ is a matrix $D - B - B^T$ without i-th row and i-th column.

Definition

A $G = (V, E)$ *be a graph (or* $\overrightarrow{G} = (V, E)$ *be a digraph) is called weighted graph (digraph) if and only if each edge* $e_i \in E$ *has an associated some positive number* $w(e_i)$ *which is called weight (cost, length).*

Weight of the spanning tree is a sum of the weights of its edges.

Kruskal's algorithm
Input: Connected weighted graph $G = (V, E)$.
Output: Minimum spanning tree T.
Suppose the graph has n vertices. We will sort the weights of edges in non-decreasing order. We start with the discrete factor T of the given graph. In each iteration, we add the edge with the smallest weight to T so that no cycle is created. If T has $n - 1$ edges, then we finish and T is the minimum spanning tree of the graph G.

If the graph has m edges, algorithm complexity is $O(m \cdot \log n)$.

The Kruskal's algorithm is a graph theory algorithm used to find the minimum spanning tree for a given graph. The algorithm works by starting with all the vertices in the graph and connecting them to form a tree. The tree is then trimmed by removing the edges that are not part of the minimum spanning tree.

Applications of Kruskal's algorithm

- Designing rail and road networks to connect several cities
- Placing microwave towers
- Designing irrigation channels
- Designing fiber-optic grids

Distance in weighted graph
Positive number $w(\{i,j\})$ is the weight of the edge $\{i,j\}$. The length of (u,v) path in the weighted graph is the sum of the weights of the edges of this path. The shortest path between two vertices is the path with the shortest length between these vertices. **Distance of two vertices v_i a v_j in the weighted graph**, denoted $d_w(v_i, v_j)$, is the length of the shortest path from v_i to v_j.

In the following Dijkstra's algorithm, we initially have given two vertices, let's denote them a, z, whose distance we want to calculate. We assign $L(v_i)$ labels to vertices v_i, which are temporary at first, subject to change, and later become permanent. If the label $L(v_i)$ is permanent for the vertex v_i, then the value of $L(v_i)$ is the length of the shortest path from the vertex a to the vertex v_i.

Dijkstra's algorithm

Input: Connected weighted graph $G = (V, H)$, vertices a, z.

Output: $L(z)$ is the length of the shortest path from the vertex a to the vertex z.

1. Assume $L(a) = 0$. For all vertices $x \neq a$, let $L(x) = \infty$.
2. If $z \notin V$, then we finish and $L(z)$ is the length of the shortest path from a to z.
3. Let's choose the vertex $v \in V$ with the smallest value of $L(v)$. The set $V = V - \{v\}$.
4. We assign the label $L(x) = \min\{L(x), L(v) + w(\{v, x\})$ to each vertex $x \in V$ that is adjacent to the vertex v. Jump to step 2.

Complexity algorithm is $O(n^2)$.

Dijkstra's algorithm

Dijkstra's graph search algorithm finds the shortest path between two nodes in a graph. It is an iterative algorithm that starts with the source node and works its way to the destination node. For each new node discovered, Dijkstra's algorithm calculates the shortest path to the destination node using the currently known distances. When traversing using Dijkstra's algorithm, any node in the graph can be considered the root node.

Applications of Dijkstra's algorithm

- Dijkstra's algorithm is used in network routing protocols to calculate the best route between two nodes.
- It is used in algorithms for solving the shortest path problem, such as the A* algorithm. The Bellman-Ford algorithm uses Dijkstra's algorithm to find the shortest path from a source node to all other nodes in a graph.
- Dijkstra's algorithm is used in many artificial intelligence applications, such as game playing and search engines.
- Maps - Finding the shortest and/or most affordable route for a car from one city to another.
- Satellite navigation systems - to show drivers the fastest path they can drive from one point in a city to the other.

Distance in weighted digraph

Similar to weighted graphs, we also define the distance between two vertices in weighted digraphs. We just have to consider the given orientation of the edges. Let the positive number $w((i,j))$ be the weight of the edge (i,j). A weighted digraph can be described by a cost matrix.

Definition

Let $\overrightarrow{G} = (V, E)$ *be a digraph, where* $V = \{v_1, v_2, \cdots, v_n\}$. **Digraph cost matrix** *is a* $n \times n$ *matrix* $W = (w_{ij})$, *where*

$$w_{ij} = \begin{cases} w((v_i, v_j)) & ak\ (v_i, v_j) \in E, \\ \infty & ak\ (v_i, v_j) \notin E, \\ 0 & ak\ i = j. \end{cases}$$

The length of directed path in a weighted digraph is the sum of the weights of the edges of this directed path. The shortest directed path from vertex v_i to vertex v_j is the directed path with the shortest length among all paths from v_i to v_j. **The distance between two vertices v_i and v_j in the weighted digraph**, denoted by $\overrightarrow{d}_w(v_i, v_j)$, is the length of the shortest directed path from v_i to v_j. If no such path exists, then $\overrightarrow{d}_w(v_i, v_j) = \infty$. It is obvious that $\overrightarrow{d}_w(v_i, v_i) = 0$. To determine the distances of two vertices in a weighted digraph, we can use the previous Dijkstra's algorithm, if we consider the direction of the edges.

We want to find out the distances between all pairs of vertices, so we calculate the distance matrix.

Definition

Let $\overrightarrow{G} = (V, H)$ be a digraph on n vertices. **Distance matrix** *for given digraph $\overrightarrow{G}$ is $n \times n$ matrix $D = (d_{ij})$, where $d_{ij} = \overrightarrow{d}_w(v_i, v_j)$.*

To calculate the distance matrix in the weighted digraph, we use Floyd's algorithm.

Floyd's algorithm

Input: Weighted digraph with vertices $v_1, v_2, \dots v_n$.

Output: Distance matrix $D = (d_{ij})$.

1. We put $D^{(0)} = W$.
2. We create a matrix $D^{(k)} = (d_{ij}^{(k)})$ such that $d_{ij}^{(k)} = \min\{d_{ij}^{(k-1)}, d_{ik}^{(k-1)} + d_{kj}^{(k-1)}\}$
3. If $k = n$, we finish and the matrix $D^{(k)} = D$. If $k < n$, we put $k = k + 1$ and jump to step 2.

Complexity algorithm is $O(n^3)$.

The Floyd's algorithm is a graph theory algorithm used to find the shortest path between all pairs of vertices in a graph. The algorithm works by constructing a table of shortest paths from each vertex to every other vertex in the graph. Like the Dijkstra's algorithm, it calculates the shortest path in a digraph. However, unlike the algorithm that use only one source to compute the shortest distance, the Floyd's algorithm calculates the shortest distances for all pairs of vertices in a digraph.

Applications of Floyd's algorithm

- Computer science - Floyd's can be used to find the best path between two vertices in a graph.
- Networks - Is used to find the shortest path between two points in a network.
- Optimal routing - Finds the path with the maximum flow between two vertices.
- Pathfinder networks - Can be used for fast computations of Pathfinder networks.