Basic concepts.
Degree sequences.

Graph theory
for MSc students in Computer Science

University of Szeged

Szeged, 2024.

Course homepage. `https://www.math.u-szeged.hu/~ngaba/graph/`

Course homepage. `https://www.math.u-szeged.hu/~ngaba/graph/`

COURSE REQUIREMENTS

Practice. There will be **2 practice tests** (25+25 points): on 22 October and on 10 December. (One of them can be retaken in the first week of the exam period.) + **bonus points**.

Lecture. During the exam period, **written exams** will be announced to test the students' knowledge. + **Bonus points** can be earned during the semester.

Course homepage. https://www.math.u-szeged.hu/~ngaba/graph/

COURSE REQUIREMENTS

Practice. There will be **2 practice tests** (25+25 points): on 22 October and on 10 December. (One of them can be retaken in the first week of the exam period.) + **bonus points**.

Lecture. During the exam period, **written exams** will be announced to test the students' knowledge. + **Bonus points** can be earned during the semester.

Grades for both courses:

0 – 50%: fail (1)
51 – 62%: pass (2)
63 – 75%: satisfactory (3)
76 – 87%: good (4)
88 – 100%: excellent (5).

Course homepage. https://www.math.u-szeged.hu/~ngaba/graph/

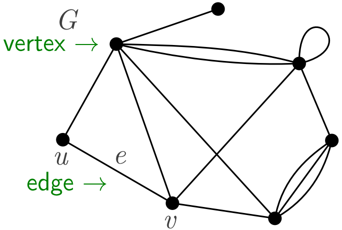
COURSE REQUIREMENTS

Practice. There will be **2 practice tests** (25+25 points): on 22 October and on 10 December. (One of them can be retaken in the first week of the exam period.) + **bonus points**.

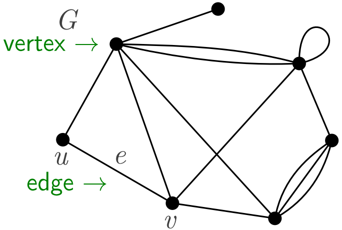
Lecture. During the exam period, **written exams** will be announced to test the students' knowledge. + **Bonus points** can be earned during the semester.

TEXTBOOK for this course

Csaba, Hajnal, Nagy: *Graph theory for MSc students in computer science*
http://eta.bibl.u-szeged.hu/2479/

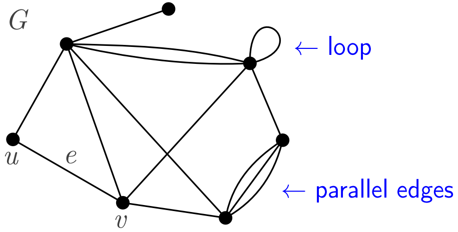


Informal definition. A multigraph G consists of a finite number of vertices and edges such that each edge is incident to exactly one or two vertices.



Informal definition. A multigraph G consists of a finite number of vertices and edges such that each edge is incident to exactly one or two vertices.

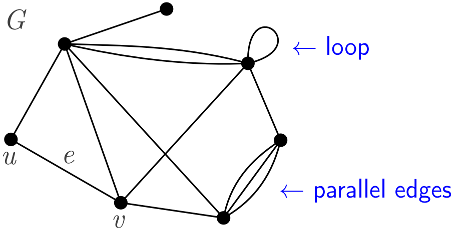
The set of vertices of G is denoted by $V(G)$ or just V, and it is called the vertex set of G. The set of edges of G is denoted by $E(G)$ or just E, and it is called the edge set of G. (Formally, V and E can be arbitrary finite sets.)



Informal definition. A multigraph G consists of a finite number of vertices and edges such that each edge is incident to exactly one or two vertices.

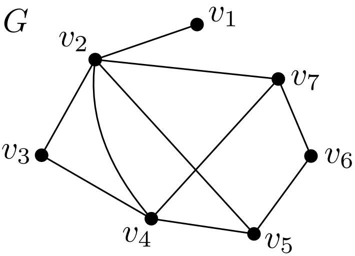
The set of vertices of G is denoted by $V(G)$ or just V, and it is called the vertex set of G. The set of edges of G is denoted by $E(G)$ or just E, and it is called the edge set of G. (Formally, V and E can be arbitrary finite sets.)

Def. An edge e is a loop, if it is incident to only one vertex (i.e. "the endpoints of e coincide"). The edges e and f are parallel edges (or multiple edges), if they are incident to the same two vertices.



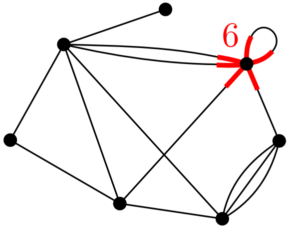
Informal definition. A multigraph G consists of a finite number of vertices and edges such that each edge is incident to exactly one or two vertices.

Terminology. If an edge e is incident to the vertices u and v in G, then we will say that "the edge e connects the vertices u and v", "e is an edge between u and v" or "u and v are the endpoints of e", etc. Two vertices are called adjacent, if they are connected by an edge. We say that v is a neighbor of u, if u and v are adjacent vertices in the multigraph. The neighborhood of u, denoted by $N(u)$, is the set of neighbors of u.



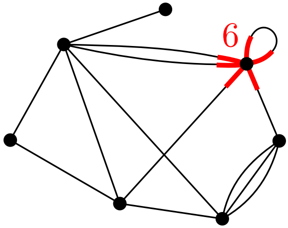
Definition A graph (or simple graph) is a multigraph that contains no loops or parallel edges.

Degree.



In a multigraph G, the degree of a vertex v is the number of edges incident to v, where loops are counted twice. The degree of v is denoted by $\deg(v)$ or $\deg_G(v)$.

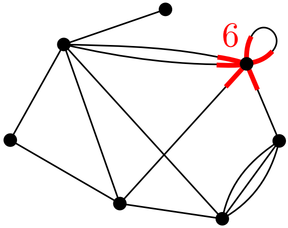
Degree.



In a multigraph G, the degree of a vertex v is the number of edges incident to v, where loops are counted twice. The degree of v is denoted by $\deg(v)$ or $\deg_G(v)$.

Note. In a (simple) graph G, the degree of vertex v is just the number of neighbors of v. (This is not necessarily true in multigraphs.)

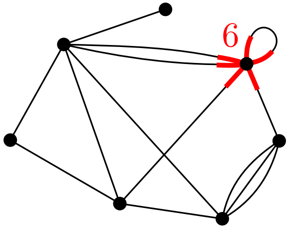
Degree.



In a multigraph G, the **degree** of a vertex v is the number of edges incident to v, where loops are counted twice. The degree of v is denoted by $\deg(v)$ or $\deg_G(v)$.

Def. A vertex is called **isolated** if its degree is 0, i.e. if there are no edges incident to it. 

Degree.



In a multigraph G, the degree of a vertex v is the number of edges incident to v, where loops are counted twice. The degree of v is denoted by $\deg(v)$ or $\deg_G(v)$.

Def. A vertex is called isolated if its degree is 0, i.e. if there are no edges incident to it.

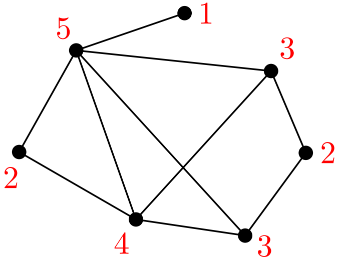
Def. A multigraph is called regular if all of its vertices have the same degree. If the common degree is d in a regular multigraph G, then we also say that G is d-regular.

Handshake lemma. In any multigraph G, the sum of the degrees of all vertices of G is equal to twice the number of edges of G.

With formulas: $$\sum_{v \in V(G)} \deg(v) = 2 \cdot |E(G)|\,,$$ for any multigraph G.

Handshake lemma. In any multigraph G, the sum of the degrees of all vertices of G is equal to twice the number of edges of G.

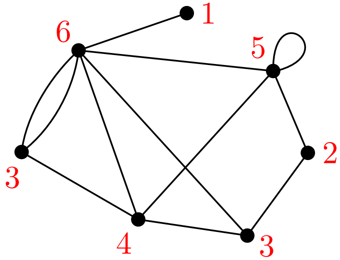
With formulas: $\sum_{v \in V(G)} \deg(v) = 2 \cdot |E(G)|$, for any multigraph G.



$$1 + 2 + 2 + 3 + 3 + 4 + 5 = 2 \cdot 10$$

Handshake lemma. In any multigraph G, the sum of the degrees of all vertices of G is equal to twice the number of edges of G.

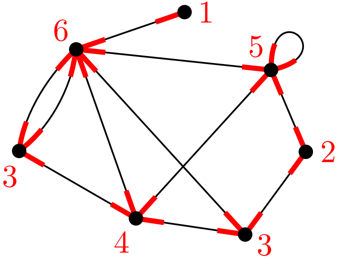
With formulas: $\displaystyle\sum_{v\in V(G)} \deg(v) = 2\cdot|E(G)|\,,$ for any multigraph G.



$$1+2+3+3+4+5+6=2\cdot 12$$

Handshake lemma. In any multigraph G, the sum of the degrees of all vertices of G is equal to twice the number of edges of G.

With formulas: $\sum_{v \in V(G)} \deg(v) = 2 \cdot |E(G)|$, for any multigraph G.

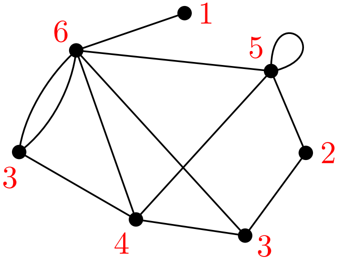


$$1 + 2 + 3 + 3 + 4 + 5 + 6 = 2 \cdot 12$$

Proof. Both sides of the equation count the total number of 'end segments' of edges in G. □

Handshake lemma. In any multigraph G, the sum of the degrees of all vertices of G is equal to twice the number of edges of G.

With formulas: $\sum_{v \in V(G)} \deg(v) = 2 \cdot |E(G)|$, for any multigraph G.

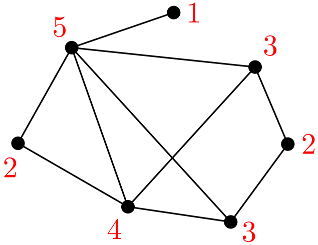


Corollary. The sum of the degrees of all vertices is even in any multigraph.

In other words. The number of vertices with odd degree is always even.

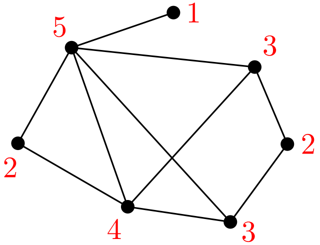
Def. The degree sequence of a multigraph is the sequence of degrees of all its vertices, sorted in decreasing (nonincreasing) order.

For example, the degree sequence of the graph in the figure is $5, 4, 3, 3, 2, 2, 1$.



Def. The degree sequence of a multigraph is the sequence of degrees of all its vertices, sorted in decreasing (nonincreasing) order.

For example, the degree sequence of the graph in the figure is $5, 4, 3, 3, 2, 2, 1$.

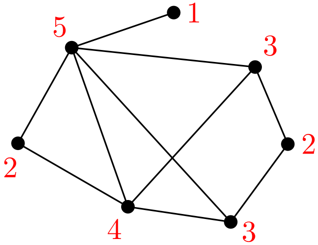


Graph realization problem. For a given input sequence $\mathbf{d}$ of nonnegative integers, decide whether there exists a graph whose degree sequence is $\mathbf{d}$. (If such a graph G exists, we say that that G realizes $\mathbf{d}$.)

For example, the sequence $5, 4, 3, 3, 2, 2, 1$ can be realized by a graph (e.g., the graph in the figure above is a realization).

Def. The degree sequence of a multigraph is the sequence of degrees of all its vertices, sorted in decreasing (nonincreasing) order.

For example, the degree sequence of the graph in the figure is $5, 4, 3, 3, 2, 2, 1$.



Graph realization problem. For a given input sequence $\mathbf{d}$ of nonnegative integers, decide whether there exists a graph whose degree sequence is $\mathbf{d}$. (If such a graph G exists, we say that that G realizes $\mathbf{d}$.)

Remark. The multigraph (etc.) realization problem is defined analogously.

Realization by multigraphs.
The decreasing sequence $d_1, d_2, \ldots, d_n$ of nonnegative integers can be realized by multigraph if and only if the sum $d_1 + d_2 + \cdots + d_n$ is even.

Proof. Easy. See Proposition 2.1 in the textbook. $\square$

Realization by multigraphs.

The decreasing sequence $d_1, d_2, \ldots, d_n$ of nonnegative integers can be realized by multigraph if and only if the sum $d_1 + d_2 + \cdots + d_n$ is even.

Proof. Easy. See Proposition 2.1 in the textbook. $\square$

Realization by loopless multigraphs.

The *decreasing* sequence $d_1, d_2, \ldots, d_n$ of nonnegative integers can be realized by loopless multigraph if and only if

- $d_1 + d_2 + \cdots + d_n$ is even, AND
- $d_1 \leq d_2 + d_3 + \cdots + d_n$.

(Note that d_1 is a largest element in the sequence, by the decreasing order.)

We omit the proof.

The (simple) graph realization problem is more difficult. Instead of giving an explicit description of the degree sequences, we present an algorithm instead. First we need to define the required operation on sequences:

The (simple) graph realization problem is more difficult. Instead of giving an explicit description of the degree sequences, we present an algorithm instead. First we need to define the required operation on sequences:

Havel–Hakimi operation. A decreasing(!) sequence $d_1, \ldots, d_n$ of nonnegative integers is given (where $n \geq 2$); this sequence is denoted by $\mathbf{d}$. The sequence HH($\mathbf{d}$) is obtained from $\mathbf{d}$ by

- removing the first element (d_1) from $\mathbf{d}$, and then
- decreasing the first d_1 elements of the obtained sequence $d_2, \ldots, d_n$ by 1.

(If $d_1 > n - 1$, then HH($\mathbf{d}$) is not defined.)

The (simple) graph realization problem is more difficult. Instead of giving an explicit description of the degree sequences, we present an algorithm instead. First we need to define the required operation on sequences:

Havel–Hakimi operation. A decreasing(!) sequence $d_1, \ldots, d_n$ of nonnegative integers is given (where $n \geq 2$); this sequence is denoted by $\mathbf{d}$. The sequence HH($\mathbf{d}$) is obtained from $\mathbf{d}$ by

- removing the first element (d_1) from $\mathbf{d}$, and then
- decreasing the first d_1 elements of the obtained sequence $d_2, \ldots, d_n$ by 1.

(If $d_1 > n - 1$, then HH($\mathbf{d}$) is not defined.)

Examples.

$5, 4, 3, 3, 3, 3, 2, 1 \;\rightsquigarrow\; 3, 2, 2, 2, 2, 2, 1$

$3, 2, 2, 2, 2, 2, 1 \;\rightsquigarrow\; 1, 1, 1, 2, 2, 1$

$6, 4, 4, 3, 3, 1 \;\rightsquigarrow\;$ not defined.

Havel–Hakimi lemma. The decreasing(!) sequence $\mathbf{d}$ of n nonnegative integers (where $n \geq 2$) can be realized by (simple) graph, if and only if

- $d_1 \leq n - 1$ AND
- the sequence HH$(\mathbf{d})$, after reordering it in decreasing order, can be realized by (simple) graph.

Havel–Hakimi lemma. The decreasing(!) sequence $\mathbf{d}$ of n nonnegative integers (where $n \geq 2$) can be realized by (simple) graph, if and only if

- $d_1 \leq n - 1$ AND
- the sequence $\mathsf{HH}(\mathbf{d})$, after reordering it in decreasing order, can be realized by (simple) graph.

Proof. See Lemma 2.3 in the textbook. $\square$

Havel–Hakimi lemma. The decreasing(!) sequence $\mathbf{d}$ of n nonnegative integers (where $n \geq 2$) can be realized by (simple) graph, if and only if

- $d_1 \leq n - 1$ AND
- the sequence HH$(\mathbf{d})$, after reordering it in decreasing order, can be realized by (simple) graph.

Proof. See Lemma 2.3 in the textbook. $\square$

Examples. The sequence $5, 4, 3, 3, 3, 3, 2, 1$ can be realized by simple graph $\iff$ The sequence $3, 2, 2, 2, 2, 2, 1$ can be realized by simple graph.

Havel–Hakimi lemma. The decreasing(!) sequence $\mathbf{d}$ of n nonnegative integers (where $n \geq 2$) can be realized by (simple) graph, if and only if

- $d_1 \leq n-1$ AND
- the sequence $\mathsf{HH}(\mathbf{d})$, after reordering it in decreasing order, can be realized by (simple) graph.

Proof. See Lemma 2.3 in the textbook. $\square$

Examples. The sequence $5, 4, 3, 3, 3, 3, 2, 1$ can be realized by simple graph $\iff$ The sequence $3, 2, 2, 2, 2, 2, 1$ can be realized by simple graph.

The sequence $3, 2, 2, 2, 2, 2, 1$ can be realized by simple graph. $\iff$ The (reordered) sequence $2, 2, 1, 1, 1, 1$ can be realized by simple graph.

Havel–Hakimi lemma. The decreasing(!) sequence $\mathbf{d}$ of n nonnegative integers (where $n \geq 2$) can be realized by (simple) graph, if and only if

- $d_1 \leq n-1$ AND
- the sequence $\mathrm{HH}(\mathbf{d})$, after reordering it in decreasing order, can be realized by (simple) graph.

Proof. See Lemma 2.3 in the textbook. $\square$

Examples. The sequence $5,4,3,3,3,3,2,1$ can be realized by simple graph $\iff$ The sequence $3,2,2,2,2,2,1$ can be realized by simple graph.

The sequence $3,2,2,2,2,2,1$ can be realized by simple graph. $\iff$ The (reordered) sequence $2,2,1,1,1,1$ can be realized by simple graph.

$\vdots$

This can be iterated: We found an ALGORITHM!

For a given decreasing input sequence $\mathbf{d}$ of nonnegative integers, the following algorithm decides whether $\mathbf{d}$ can be realized by simple graph or not.

Havel–Hakimi algorithm (on input sequence $\mathbf{d}$).

- If $\mathbf{d}$ is a one-element sequence, then it can be realized by simple graph if and only if it is the sequence 0, and the algorithm terminates.
- If the first element of $\mathbf{d}$ is greater than or equal to the number of elements of $\mathbf{d}$, then $\mathbf{d}$ cannot be realized and the algorithm terminates.
- Otherwise, calculate HH$(\mathbf{d})$.
- If the sequence HH$(\mathbf{d})$ contains a negative number, then $\mathbf{d}$ cannot be realized and the algorithm terminates.
- Otherwise, reorder the sequence HH$(\mathbf{d})$ in decreasing order, and invoke this algorithm recursively on input sequence HH$(\mathbf{d})$. The obtained answer is the answer to the initial question (on input $\mathbf{d}$).

For a given decreasing input sequence $\mathbf{d}$ of nonnegative integers, the following algorithm decides whether $\mathbf{d}$ can be realized by simple graph or not.

Havel–Hakimi algorithm (on input sequence $\mathbf{d}$).

- If $\mathbf{d}$ is a one-element sequence, then it can be realized by simple graph if and only if it is the sequence 0, and the algorithm terminates.
- If the first element of $\mathbf{d}$ is greater than or equal to the number of elements of $\mathbf{d}$, then $\mathbf{d}$ cannot be realized and the algorithm terminates.
- Otherwise, calculate HH$(\mathbf{d})$.
- If the sequence HH$(\mathbf{d})$ contains a negative number, then $\mathbf{d}$ cannot be realized and the algorithm terminates.
- Otherwise, reorder the sequence HH$(\mathbf{d})$ in decreasing order, and invoke this algorithm recursively on input sequence HH$(\mathbf{d})$. The obtained answer is the answer to the initial question (on input $\mathbf{d}$).

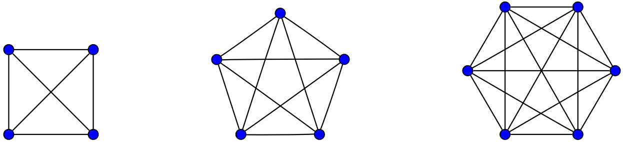
Examples. See the two examples after Theorem 2.5 in the textbook.

Havel–Hakimi algorithm (on input sequence $\mathbf{d}$).

- If $\mathbf{d}$ is a one-element sequence, then it can be realized by simple graph if and only if it is the sequence 0, and the algorithm terminates.
- If the first element of $\mathbf{d}$ is greater than or equal to the number of elements of $\mathbf{d}$, then $\mathbf{d}$ cannot be realized and the algorithm terminates.
- Otherwise, calculate HH$(\mathbf{d})$.
- If the sequence HH$(\mathbf{d})$ contains a negative number, then $\mathbf{d}$ cannot be realized and the algorithm terminates.
- Otherwise, reorder the sequence HH$(\mathbf{d})$ in decreasing order, and invoke this algorithm recursively on input sequence HH$(\mathbf{d})$. The obtained answer is the answer to the initial question (on input $\mathbf{d}$).

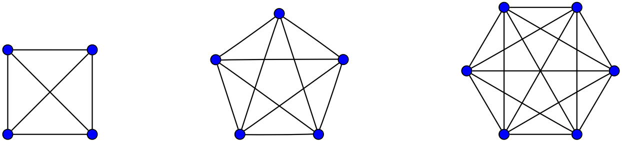
Remark. This recursive algorithm can be terminated with return value "YES" at any point when the actual sequence can be trivially realized by simple graph. (In practice, the algorithm returns "YES" if it reaches the constant sequence $0, 0, \ldots, 0$.)

Definition. A complete graph is a graph in which every pair of distinct vertices is connected by an edge. The complete graph on n vertices is denoted by K_n.



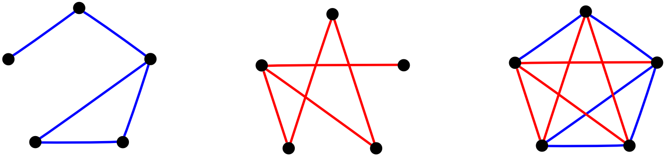
K_4 K_5 K_6

Definition. A complete graph is a graph in which every pair of distinct vertices is connected by an edge. The complete graph on n vertices is denoted by K_n.



K_4 K_5 K_6

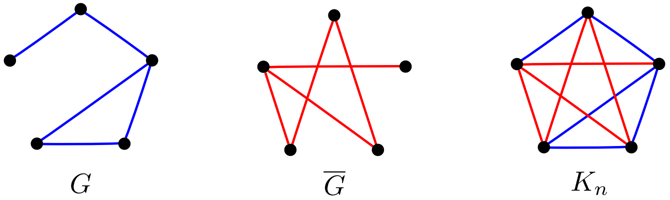
Definition. The complement of a graph G, denoted by $\overline{G}$, is a simple graph on the same vertex set as G, such that any two vertices are adjacent in $\overline{G}$ if and only if they are not adjacent in G.



G $\overline{G}$ K_n

Definition. A complete graph is a graph in which every pair of distinct vertices is connected by an edge. The complete graph on n vertices is denoted by K_n.

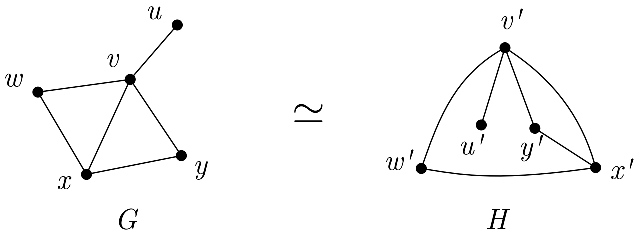
Definition. The complement of a graph G, denoted by $\overline{G}$, is a simple graph on the same vertex set as G, such that any two vertices are adjacent in $\overline{G}$ if and only if they are not adjacent in G.



Remark. $\overline{G}$ can be obtained from the complete graph on vertex set $V(G)$ by removing the edges of G.

Definition. The (simple) graphs G and H are said to be isomorphic, if there exist a bijection $\phi\colon V(G) \to V(H)$ such that any two vertices u and v are adjacent in G if and only if the vertices $\phi(u)$ and $\phi(v)$ are adjacent in H. (Such a bijection ϕ is called graph isomorphism between G and H.)

Notation. $G \simeq H$.



Informally, isomorphic graphs are "essentially the same" (thus they are considered the same in graph theory almost always), the only difference is in the "names" of vertices. We leave the students to the adopt the definition of graph isomorphism to multigraphs.