CS213/293 Data Structure and Algorithms 2025

Lecture 13: Graphs - basics

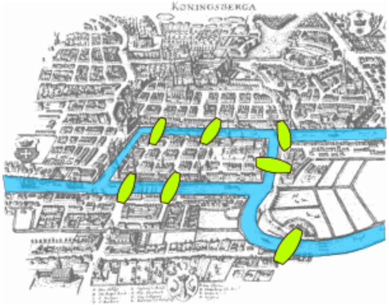
Instructor: Ashutosh Gupta

IITB India

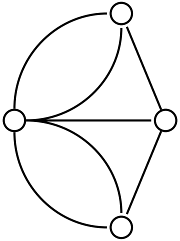
Compile date: October 7, 2025

The problem of Konigsberg's bridges

Problem: Find a walk through the city that would cross each of those bridges once and only once.



(Source: Wikipedia)

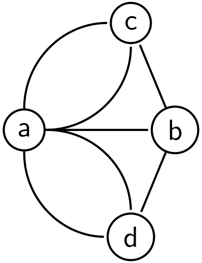


We may view the problem as visiting all nodes without repeating an edge in the above graph.

The first graph theory problem. Euler gave the solution!

Graphs

A graph consists of vertices (also known as nodes), which are connected by edges.



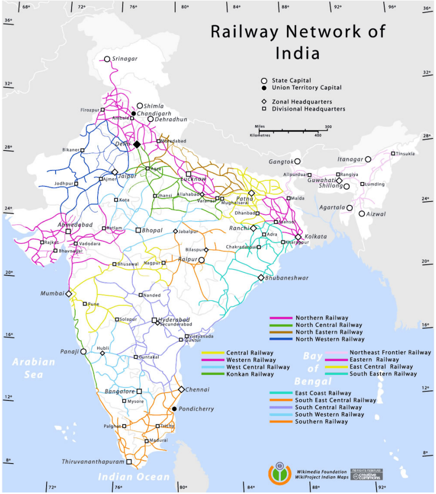
The above is a graph $G = (V, E)$, where

$V = \{a, b, c, d\}$ and

E is a multiset.

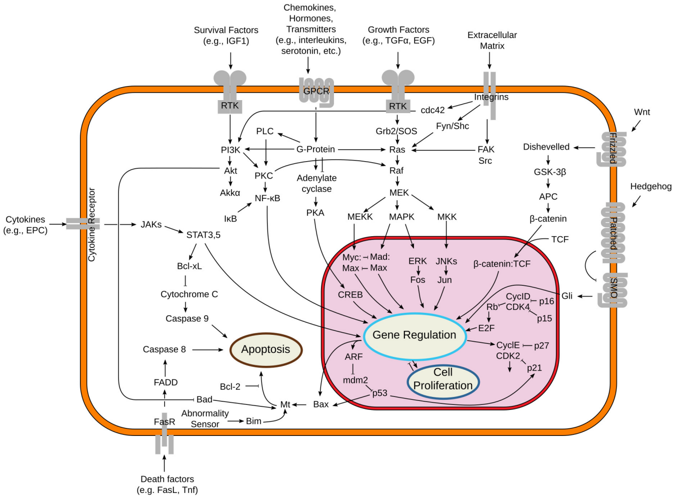
$E = \{\{a, b\}, \{a, c\}, \{a, c\}, \{a, d\}, \{a, d\}, \{b, c\}, \{b, d\}\}.$

Example: graphs are everywhere



(Source: Internet)

Example: graphs are everywhere (2)



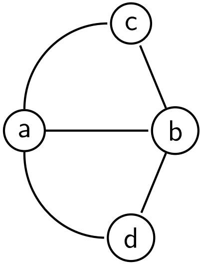
(Source: Wikipedia)

Formal definition

Definition 13.1

A graph $G = (V, E)$ consists of

- set of vertices V and
- set of edges E is a set of unordered pairs of elements of V.



Commentary: In the bridge example, E was a multiset, and here E is a set. If we want to support multiset, we can define $E \subseteq \mathit{unorderedPairs}(V) \times \mathbb{N}$, which is a natural extension of the above definition. $\mathit{unorderedPairs}(V) = \{\{a, b\} | a, b \in V \wedge a \neq b\}$

Topic 13.1

Basic Terminology

Adjacency and degree

Consider a graph $G = (V, E)$.

Definition 13.2

Let *adjacent*$(v) = \{v' | \{v, v'\} \in E\}$.

Definition 13.3

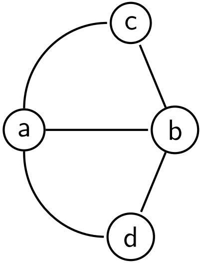
Let *degree*$(v) = |adjacent(v)|$.

Exercise 13.1

a. What is $\sum_{v \in V} degree(v)$?

b. Is $\{v, v\} \in E$ possible? Answer: No.

Example 13.1



$adjacent(a) = \{c, b, d\}$ and $adjacent(d) = \{a, b\}$.

$degree(a) = 3$ and $degree(d) = 2$.

Commentary: $\sum_{v \in V} degree(v) = 2|E|$. We are disallowing $\{v, v\} \in E$ because we consider an unordered pair to be a set of size 2. Some definitions allow $\{v, v\}$ if we view an unordered pair as a multiset.

Paths, simple paths, and cycles

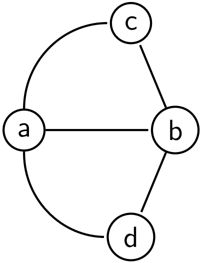
Consider a graph $G = (V, E)$.

Definition 13.4
A path is a sequence of vertices $v_1, \ldots, v_n$ such that $\{v_i, v_{i+1}\} \in E$ for each $i \in [1, n)$.

Definition 13.5
A simple path is a path $v_1, \ldots, v_n$ such that $v_i \neq v_j$ for each $i < j \in [1, n]$.

Definition 13.6
A cycle is a path $v_1, \ldots, v_n$ such that $v_1, \ldots, v_{n-1}$ is a simple path and $v_1 = v_n$.

Example 13.2



abcad is a path but not a simple path.
abd is a simple path.
abda is a cycle.

Exercise 13.2
a. Can there be an empty path?
b. Is *b* a cycle?

Subgraph

Consider a graph $G = (V, E)$.

Definition 13.7

A graph $G' = (V', E')$ is a subgraph of G if $V' \subseteq V$ and $E' \subseteq E$.

Definition 13.8

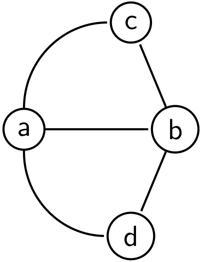
For a set of vertices V', let $G - V'$ be $(V - V', \{e | e \in E \land e \subseteq V - V'\})$.

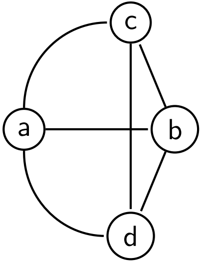
Definition 13.9

For a set of edges E', let $G - E'$ be $(V \cap \bigcup(E - E'), E - E')$.

Example 13.3

The left graph is a subgraph of the right graph.





Connected graph

Consider a graph $G = (V, E)$.

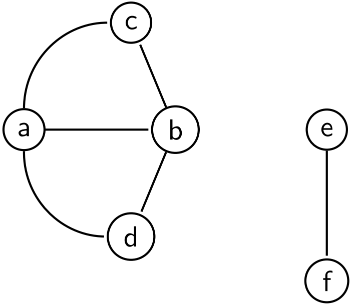
Definition 13.10

G is connected if for each $v, v' \in V$, there is a path $v, \dots, v'$ in E.

Definition 13.11

A graph G' is a connected component of G if G' is a maximal connected subgraph of G.

Example 13.4



The above is not a connected graph.

The above has two connected components.

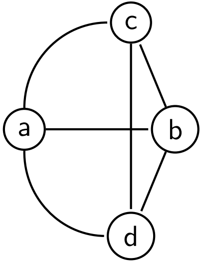
Complete graph

Consider a graph $G = (V, E)$.

Definition 13.12

G is a complete graph if for all distinct pairs $v_1, v_2 \in V$, $v_1 \in adjacent(v_2)$.

Example 13.5



Exercise 13.3

If $|V| = n$, how many edges does a complete graph have?

Topic 13.2

Tree (a new non-recursive definition of tree)

Tree

Consider a graph $G = (V, E)$.

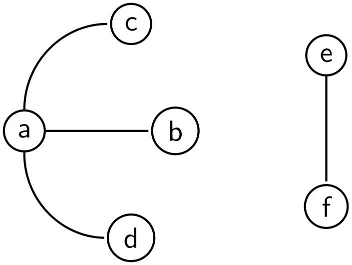
Definition 13.13
G is a tree if G is connected and has no cycles.

Definition 13.14
G is a forest if G is a disjoint union of trees.

Definition 13.15
$G = (V, E, v)$ is a rooted tree if (V, E) is a tree and $v \in V$ is called the root.

The trees in the earlier lectures are rooted trees.

Example 13.6



The above is a forest containing two trees.

Exercise 13.4
Which nodes of a tree can be the root?

Every tree has a leaf

Definition 13.16

For a tree $G = (V, E)$, a node $v \in V$ is a leaf if $degree(v) \leq 1$.

Theorem 13.1

For a finite tree $G = (V, E)$ and $|V| > 1$, there is $v \in V$ such that $degree(v) = 1$.

Proof.

Since there are no cycles in G and G is finite, there is a simple path $v_1, \ldots, v_n$ of G that cannot be extended at either end.

Therefore, there must be two nodes such that $degree(v) = 1$.

□

Number of edges in a tree

Theorem 13.2

For a finite tree $G = (V, E)$, $|E| = |V| - 1$.

Proof.

Base case:

Let $|V| = 2$. We have $|E| = 1$.

Induction step:

Let $|V| = n + 1$.

Consider a leaf $v \in V$ and $\{v, v'\} \in E$.

Since $degree(v) = 1$ in G, $G - \{v\}$ is a tree.

Due to the induction hypothesis, $G - \{v\}$ has $|V| - 2$ edges.

Hence proved. □

Number of edges in a tree

Theorem 13.3

Let $G = (V, E)$ be a finite graph. If $|E| < |V| - 1$, G is not connected.

Proof.

Let us suppose there are cycles in the graph.

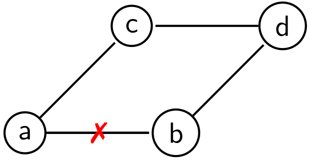
If we remove an edge from a cycle, it does not change the connectedness of any pair of vertices. (Why?)

We continue to remove such edges until no more cycles remain.

Since $|E| < |V| - 1$, the remaining graph is not a tree. Therefore, G was not connected. □

Example 13.7

Consider the following graph with a cycle



After removing $\{a, b\}$, the connectedness of a and d does not change.

Commentary: Please check the definition of a tree. The last conclusion is a direct application of the contra-positive of the definition of a tree.

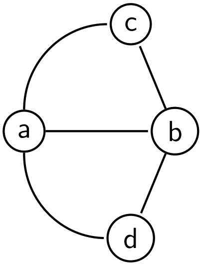
Spanning tree

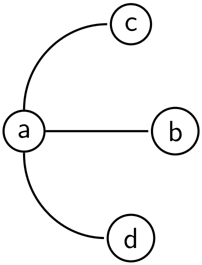
Consider a graph $G = (V, E)$.

Definition 13.17

A spanning tree of G is a subgraph of G that is a tree and contains all the vertices of G.

Example 13.8



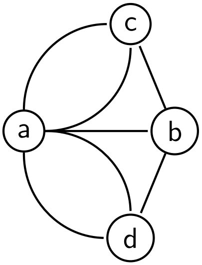


The right graph is the spanning tree of the left graph.

Topic 13.3

Multi-graph

Multi graph



Definition 13.18

A multi-graph $G = (V, E)$ consists of

- set of vertices V and
- set of edges E is a *multiset* of unordered pairs of elements of V.

The above is a graph $G = (V, E)$, where

$V = \{a, b, c, d\}$ and

$E = \{\{a, b\}, \{a, c\}, \{a, c\}, \{a, d\}, \{a, d\}, \{b, c\}, \{b, d\}\}$.

Eulerian tour

Consider a graph $G = (V, E)$.

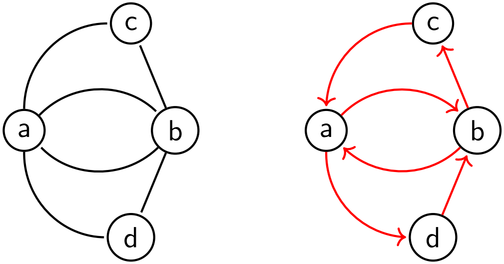
Definition 13.19

For a (multi)graph G, an Eulerian tour is a path that traverses every edge exactly once and returns to the same node.

Exercise 13.5

Why is an Eulerian tour not a cycle?

Example 13.9



Eulerean path: cadbabc

Conditions for an Eulerian tour

Theorem 13.4
A connected graph has an Eulerian tour if and only if all vertices have even degrees.

Proof.
Consider a connected graph $G = (V, E)$.We do induction over the number of edges in the graph.
Base case:
$G = (\emptyset, \emptyset)$ has an Eulerian tour. $G = (\{v_1, v_2\}, \{\{v_1, v_2\}\})$ does not have an Eulerian tour.
If $|E| < 2$, G has an Eulerian tour $\Leftrightarrow$ all vertices of G have even degrees. $\square$

Conditions for an Eulerian tour (continued)

Proof.

Induction step:

Assume $|E| \geq 2$. Let $\{\{v_1, v_2\}, \{v_2, v_3\}\} \subseteq E$.

Let us define G' using one of the following operations.

- $v_1 = v_3$: Let $G' = G - \{\{v_1, v_2\}, \{v_2, v_3\}\}$.
- $v_1 \neq v_3$: Let $G' = G + \{\{v_1, v_3\}\} - \{\{v_1, v_2\}, \{v_2, v_3\}\}$.

Let G'_i be the connected components of G'.

G has an Eulerian tour $\Leftrightarrow$ all of G'_is have Eulerian tours.(Why?)

Due to the induction hypothesis, G'_i has an Eulerian tour $\Leftrightarrow$ all vertices of G'_i have even degrees.

All vertices of G' have even degrees $\Leftrightarrow$ all vertices of G have even degrees.(Why?)

Therefore, G has an Eulerian tour $\Leftrightarrow$ all vertices of G have even degrees. □

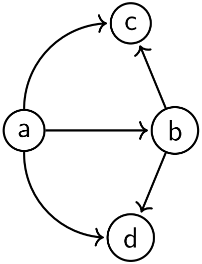
Exercise 13.6

Write the formal argument for the two "whys" in the above proof?

Topic 13.4

Directed graph

Directed graph



Definition 13.20

A graph $G = (V, E)$ consists of

- set of vertices V and
- set of edges $E \subseteq V \times V$.

The above is a directed graph $G = (V, E)$, where

$V = \{a, b, c, d\}$ and

$E = \{(a, b), (a, c), (a, d), (b, c), (b, d)\}$.

There is a path from a to d, but not d to a.

Definition 13.21

A path is a sequence of vertices $v_1, \ldots, v_n$ such that $(v_i, v_{i+1}) \in E$ for each $i \in [1, n)$.

Strongly connected component (SCC)

Consider a directed graph $G = (V, E)$.

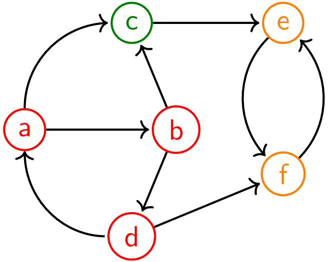
Definition 13.22

G is strongly connected if for each $v, v' \in V$, there is a path $v, \dots, v'$ in E.

Definition 13.23

A graph G' is a strongly connected component (SCC) of G if G' is a maximal strongly connected subgraph of G.

Example 13.10



abd, c, and ef are SCCs.

SCC-Graph

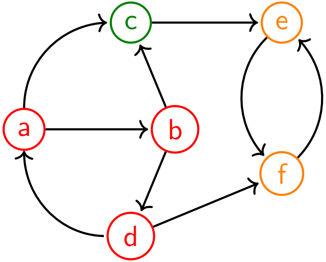
Let G be a directed graph.

Definition 13.24

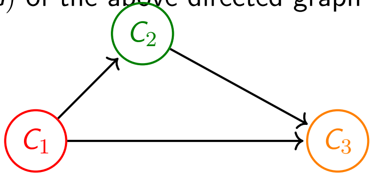
SCC-graph $SCC(G)$ is defined as follows.

- Let $C_1, \ldots, C_n$ be SCCs of G.
- For each C_i, create a vertex v_i in $SCC(G)$.
- Add an edge (v_i, v_j) to $SCC(G)$, if there are two vertices u_i and u_j in G with $u_i \in C_i, u_j \in C_j$, and $(u_i, u_j) \in E$.

Example 13.11



$SCC(G)$ of the above directed graph G is



SCC(G) is acyclic

Theorem 13.5

For any directed graph $G = (V, E)$, $SCC(G)$ is acyclic.

Proof.

Let us suppose there is a cycle in $SCC(G) = (V', E')$.

There must be $u, u' \in V'$ such that there are paths from u to u' and in the reverse direction.

Let C and C' be the SSCs in G corresponding to u and u', respectively.

There must be a path from nodes in C to nodes in C' and in the reverse direction.

C and C' cannot be SSCs of G. Contradiction. $\square$

Topic 13.5

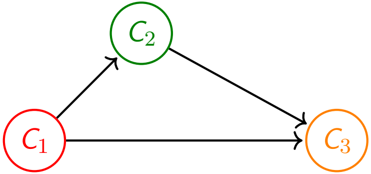
Directed acyclic graph (DAG)

Directed acyclic graph (DAG)

Consider a directed graph $G = (V, E)$.

Definition 13.25

G is a directed acyclic graph (DAG) if G has no cycles.



The above is a directed acyclic graph.

Exercise 13.7

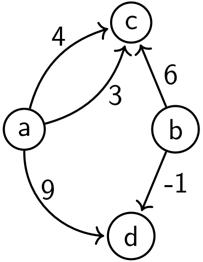
Define a tree from a DAG.

Commentary: We may view that DAG $SCC(G)$ is embedded in graph G.

Topic 13.6

Labeled graph

Directed labeled graph



Definition 13.26

A graph $G = (V, E)$ consists of

- set of vertices V and
- set of edges $E \subseteq V \times L \times V$,

where L is the set of labels.

The above is a labelled directed graph $G = (V, E)$, where

$L = \mathbb{Z}$, $V = \{a, b, c, d\}$ and

$E = \{(a, 3, c), (a, 4, c), (a, 9, d), (b, 6, c), (b, -1, d)\}$.

Topic 13.7

Representation of a graph

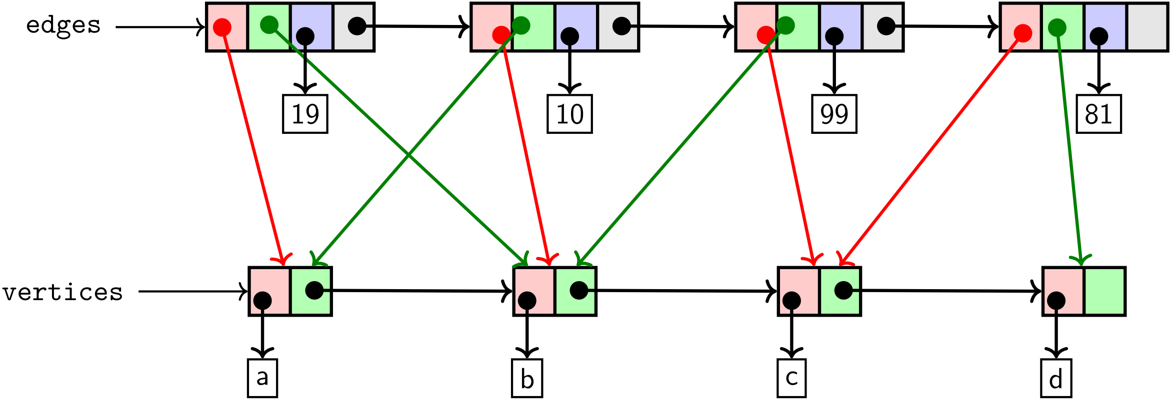
Representations of a graph

- Edge list
- Adjacency list
- Matrix

Edge list

- Store vertices as a sequence (array/list)
- Store edges as a sequence with pointers to vertices

Example: edge list



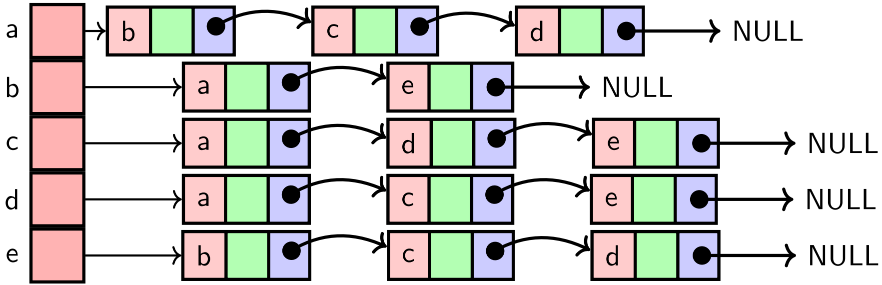
Exercise 13.8

a. What is the cost of computing *adjacent*(v)?

b. What is the cost of insertion of an edge?

Adjacency list

- Each vertex maintains the list of adjacent nodes.



Space: $O(|V| + \sum degree(v)) = O(|V| + |E|)$

Exercise 13.9

a. Draw the graph for the above data structure.

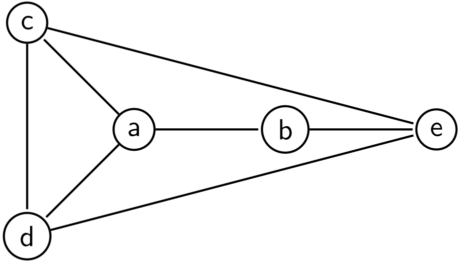
b. What is the cost of $adjacent(v)$, and find the vertices of an edge given by edge number?

c. How can we mix the edge list and adjacency list to make the above operations efficient?

> **Commentary:** The matrix is stored on a multidimensional array. If we store the matrix as a vector of vectors, then it is *similar* to the adjacency list storage of the graph instead of the adjacency matrix. Many graph algorithms are sequences of matrix operations over adjacency matrices. Matrix operations are not fast on vectors of vectors.

Adjacency Matrix

Store adjacency relation in a matrix.



	a	b	c	d	e
a	0	1	1	1	0
b	1	0	0	0	1
c	1	0	0	1	1
d	1	0	1	0	1
e	0	1	1	1	0

Space: $O(|V|^2)$

Exercise 13.10

a. What is the cost of adding a node? $O(n^2)$

b. What is the cost of $adjacent(v)$? $O(n)$

c. What is the cost of finding the vertices of an edge that is given as a pair of positions? $O(1)$

d. How can we mix edge list and adjacency matrix?

Topic 13.8

Tutorial problems

Exercise: modeling COVID

Exercise 13.11

The graph is an extremely useful modeling tool. Here is how a Covid tracing tool might work. Let V be the set of all persons. We say (p,q) is an edge (i) in E1 if their names appear on the same webpage, and (ii) in E2 if they have been together in a common location for more than 20 minutes. What significance do the connected components in these graphs have, and what does the BFS does? Does the second graph have epidemiological significance? If so, what? If not, how would you improve the graph structure to get a sharper epidemiological meaning?

Commentary: Source: Milind notes.

Exercise: Bipartite graphs

Definition 13.27
A graph $G = (V, E)$ is bipartite if $V = V_1 \uplus V_2$ and for all $e \in E$, $e \not\subseteq V_1$ and $e \subseteq V_2$.

Exercise 13.12
Show that a bipartite graph does not contain cycles of odd length.

Exercise: Planer graphs

Exercise 13.13

Let us take a plane paper and draw circles and infinite lines to divide the plane into various pieces. There is an edge (p,q) between two pieces if they share a common boundary of intersection (which is more than a point). Is this graph bipartite? Under what conditions is it bipartite?

Commentary: Source: Milind notes.

Exercise: Die hard puzzle

Exercise 13.14

There are three containers A, B, and C, with capacities of 5,3, and 2 liters, respectively. We begin with A has 5 liters of milk, and B and C are empty. There are no other measuring instruments. A buyer wants 4 liters of milk. Can you dispense this? Model this as a graph problem with the vertex set V as the set of configurations c=(c1,c2,c3) and an edge from c to d if d is reachable from c. Begin with (5,0,0). Is this graph directed or undirected? Is it adequate to model the question: How to dispense 4 liters?

Commentary: Source: Milind notes.

Topic 13.9

Problems

True or False

Exercise 13.15

Mark the following statements True / False and also provide justification.

1. A tree with 10 nodes has 10 edges.
2. A graph can have an infinite number of vertices.
3. For a graph G, $SCC(G)$ is acyclic.
4. The number of paths between two vertices in a graph is $O(|V|^2)$.
5. A connected graph must have at least $|V| - 1$ edges.

Exercise: Modeling a call center

Exercise 13.16

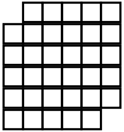
Suppose that there are M workers in a call center for a travel service that gives travel directions within a city. It provides services for N cities - C1,...,CN. Not all workers are familiar with all cities. The numbers of requests from cities per hour are R1,...,RN. A worker can handle K calls per hour. Is the number of workers sufficient to address the demand? How would you model this problem? Assume that R1,...,RN, and K are small numbers.

Commentary: Source: Milind notes.

Exercise: tiling (2023 Quiz)

Exercise 13.17

Prove that it is not possible to tile the following floor using some number of tiles shaped  .
Tiles must not be deformed and overlap.



End of Lecture 13