Connected Graphs and Spanning Trees

GAINA, Daniel

Institute of Mathematics for Industry
Kyushu University

Describing the problem I

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

1. V is a set of vertices
2. E is a (multi)set of edges

Example 1

$$\begin{tikzcd}
1 \arrow[r, dash] \arrow[d, dash] & 2 \arrow[r, dash] \arrow[d, dash] & 3 \arrow[d, dash] \\
4 \arrow[r, dash] \arrow[d, dash] & 5 \arrow[r, dash] \arrow[d, dash] & 6 \arrow[d, dash] \\
7 \arrow[r, dash] & 8 \arrow[r, dash] & 9
\end{tikzcd}$$

$V = \{1, \ldots, 9\}$
$E = \{[1,2]; [1,4]; [2,3]; [2,5]; [3,6]; [4,5]; [4,7]; [5,6]; [5,8]; [6,9]; [7,8]; [8,9]\}$

Describing the problem II

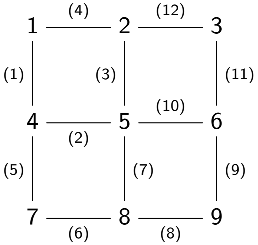
Let $G = (V, E)$ be a connected graph.

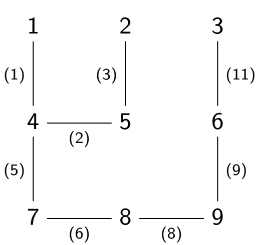
$T = (V, E')$ is spanning tree of G when

1. T is tree, and
2. $E' \subseteq E$.

Theorem 2

Every connected graph has a spanning tree.





Towards formalization

Specification

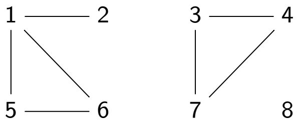
1. Define data types to represent graphs
2. Define the following functions:
   - `connected`
   - `nocycle`
   - `mktree`

Verification

1. connected(G)$\Rightarrow$ $\exists$G'$\subseteq$ G.tree(G')
2. connected(G)$\Rightarrow$ connected(mktree(G))$\wedge$ nocycle(mktree(G))

Functions on graphs

$G = (\{1,2,3,4,5,6,7,8\}, [1,2]; [1,6]; [1,5]; [3,4]; [3,7]; [4,7]; [5,6])$



- `mcc(A,G)` $=$ max. connected comp. of `A` in `G`.
  `mcc(6,G) =` $\{1,2,5,6\}$`, mcc(8,G) =` $\{8\}$
- $\#$`cc(G) =` no. of max. connected components
  $\#$`cc(G) = 3`
- `nocycle(G) = false`

Spanning forests

In the attempt of proving the desired properties we realized is much easier to prove a more general result:

Every graph has a spanning forest!

Definition 3

A **spanning forest** of a graph is a subgraph that consists of a set of spanning trees, one for each maximal connected component of the initial graph.

We define the function `mktree` which returns the spanning forest of a graph.

`G=({1,2,3,4,5,6,7,8}, [1,2];[1,6];[1,5];[3,4];[3,7];[4,7];[5,6])`

`mktree(G)=({1,2,3,4,5,6,7,8},[1,2];[1,6];[1,5];[3,4];[3,7])`

Remark 4

The value is relative to the order chosen for the edges.

Properties to prove

1. `mcc(A,G)=mcc(A,mktree(G))`
2. `#cc(G)=#cc(mktree(G))`
3. `nocycle(mktree(G))`

Then we define `connected(G):= (#cc(G)=1)` which implies

`connected(G)`$\Rightarrow$ `connected(mktree(G))`$\wedge$ `nocycle(mktree(G))`