What is another way of creating spanning trees besides guess-and-check and breadth-first search?

Theorem 5.7:

Let the depth-first search algorithm be applied to a graph $\mathcal{G}$.

(a)The edges and vertices selected form a tree.

(b) If $\mathcal{G}$ is connected, this tree is a spanning tree.

Definitions:

The tree created using the depth-first search algorithm is called a ________________________.

The edges are called ____________ and the other edges are called ____________. The labeling of the vertices is called a ________________________.

Example: Consider the graph below. Apply the depth-first search algorithm to obtain a depth-first search numbering of the vertices and use that numbering to create a spanning tree. Also list the back edges of the graph.

Example: Consider the graph below. Apply the depth-first search algorithm to obtain a depth-first search numbering of the vertices and use that numbering to create a spanning tree. Also list the back edges of the graph.

Example: Consider the graph below. Apply the (a) breadth-first search algorithm and (b) depth-first search algorithm to create a spanning tree.

Definition:

The absence of an edge, called a ____________, whose removal disconnects the graph is necessary and sufficient to guarantee that there is a way to assign directions to edges so as to produce a strongly connected directed graph.

Theorem 5.8:

Suppose depth-first search is applied to a connected graph without a bridge. If directions are assigned to tree edges by going from the lower depth-first search number to the higher and to back edges by going from the higher number to the lower, then the resulting directed graph is strongly connected.

Example: Are there any bridges in the graph below? If so, where? If not, apply the depth-first search algorithm to obtain a depth first search numbering of the vertices to assign directions to edges that will make the graph a strongly connected directed graph.

Example: Are there any bridges in the graph below? If so, where? If not, apply the depth-first search algorithm to obtain a depth first search numbering of the vertices to assign directions to edges that will make the graph a strongly connected directed graph.

Rachel Rupnow. Compiled: November 1, 2022