Contenido: Tema 1 de la Unidad 3	Fecha de entrega: 30/11/09

EJERCICIO 7

De la edición 2 del libro texto los enunciados 22.3-2 y 23-4.

22.3-2 Show how depth-first search works on the graph of Figure 22.6. Assume that the **for** loop of lines 5–7 of the DFS procedure considers the vertices in alphabetical order, and assume that each adjacency list is ordered alphabetically. Show the discovery and finishing times for each vertex, and show the classification of each edge.

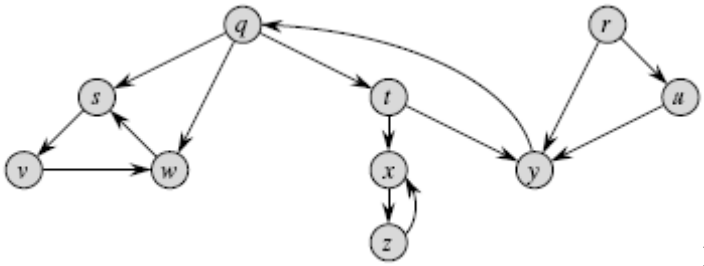

Figure 22.6

23-4 Alternative minimum-spanning-tree algorithms. In this problem, we give pseudocode for three different algorithms. Each one takes a graph as input and returns a set of edges T. For each algorithm, you must either prove that T is aminimum spanning tree or prove that T is not a minimum spanning tree. Also describe the most efficient implementation of each algorithm, whether or not it computes a minimum spanning tree.

a. MAYBE-MST-A*(G,w)*
1 sort the edges into nonincreasing order of edge weights *w*
2 $T \leftarrow E$
3 **for** each edge *e*, taken in nonincreasing order by weight
4 **do if** $T - \{e\}$ is a connected graph
5 **then** $T \leftarrow T - e$
6 **return** *T*

b. MAYBE-MST-B*(G,w)*
1 $T \leftarrow \varnothing$
2 **for** each edge *e*, taken in arbitrary order
3 **do if** $T \cup \{e\}$ has no cycles
4 **then** $T \leftarrow T \cup e$
5 **return** *T*

c. MAYBE-MST-C*(G,w)*
1 $T \leftarrow \varnothing$
2 **for** each edge *e*, taken in arbitrary order
3 **do** $T \leftarrow T \cup \{e\}$
4 **if** *T* has a cycle *c*
5 **then** let *e'* be a maximum-weight edge on *c*
6 $T \leftarrow T - \{e'\}$
7 return *T*