Consider the following distance matrix defined for the 7 species A, B, C, D, E, F, G.

	A	B	C	D	E	F	G
A	0	6	16	18	19	9	8
B	6	0	20	22	23	13	12
C	16	20	0	16	17	7	8
D	18	22	16	0	7	5	10
E	19	23	17	7	0	6	11
F	9	13	7	5	6	0	1
G	8	12	8	10	11	1	0

1. Given the distance matrix provided, construct a tree, ***T***, as follows:

    a. Begin with an empty edge list, ***E***
    b. Select the minimum-distance edge, ***e***, not in ***E***, such that the graph, ***T***, formed by the edges ***E*** $\cup$***{e}*** has no cycles. If no such edge exists, return ***T***
    c. Set ***E* = *E*** $\cup$***{e}***
    d. Goto step b

    Draw the resulting tree, ***T***.

2. Which of the seven algorithm-design techniques mentioned in Chapter 2 of the textbook does this tree-construction algorithm best illustrate? Explain your answer.

3. Given a graph with N vertices, how many times will step c in question 1 be executed?

4. A *minimum spanning tree* of a graph connects all of the graph's vertices with a minimal sum of the edge distances. Does the algorithm from question 1 achieve this objective? Justify your answer.

5. Does the algorithm given question 1 solve the Hamiltonian path problem (a minimum distance path through all vertices)? Explain your answer.

6. The following simple algorithm for computing a minimum spanning tree from a fully connected graph was given in class: *Remove the most expensive edge on each cycle, until no cycle remains*. In each iteration, this algorithm removes one edge. How many iterations will be executed in a graph with ***N*** vertices?