Exercises 3.4

1. a. Assuming that each tour can be generated in constant time, what will be the efficiency class of the exhaustive-search algorithm outlined in the text for the traveling salesman problem?

   b. If this algorithm is programmed on a computer that makes 10 billion additions per second, estimate the maximum number of cities for which the problem can be solved in (i) one hour. (ii) 24-hours. (iii) one year. (iv) one century.

2. Outline an exhaustive-search algorithm for the Hamiltonian circuit problem.

3. Outline an algorithm to determine whether a connected graph represented by its adjacency matrix has a Eulerian circuit. What is the efficiency class of your algorithm?

4. Complete the application of exhaustive search to the instance of the assignment problem started in the text.

5. Give an example of the assignment problem whose optimal solution does not include the smallest element of its cost matrix.

6. Consider the ***partition problem***: given n positive integers, partition them into two disjoint subsets with the same sum of their elements. (Of course, the problem does not always have a solution.) Design an exhaustive search algorithm for this problem. Try to minimize the number of subsets the algorithm needs to generate.

7. Consider the ***clique problem:*** given a graph G and a positive integer k, determine whether the graph contains a ***clique*** of size k, i.e., a complete subgraph of k vertices. Design an exhaustive-search algorithm for this problem.

8. Explain how exhaustive search can be applied to the sorting problem and determine the efficiency class of such an algorithm.

9. *Eight-queens problem* Consider the classic puzzle of placing eight queens on an 8×8 chessboard so that no two queens are in the same row or in the same column or on the same diagonal. How many ways are there so that

   a. no two queens are on the same square?

   b. no two queens are in the same row?

   c. no two queens are in the same row or in the same column?

Also estimate how long it would take to find all the solutions to the problem by exhaustive search based on each of these approaches on a computer capable of checking 10 billion positions per second.

10. A magic square of order n is an arrangement of the integers from 1 to n^2 in an $n \times n$ matrix, with each number occurring exactly once, so that each row, each column, and each main diagonal has the same sum.

    a. Prove that if a magic square of order n exists, the sum in question must be equal to $n(n^2 + 1)/2$.

    b. Design an exhaustive search algorithm for generating all magic squares of order n.

    c. Go to the Internet or your library and find a better algorithm for generating magic squares.

    d. Implement the two algorithms—the exhaustive search and the one you have found—and run an experiment to determine the largest value of n for which each of the algorithms is able to find a magic square of order n in less than 1 minute of your computer's time.

11. *Famous alphametic* A puzzle in which the digits in a correct mathematical expression, such as a sum, are replaced by letters is called ***cryptarithm***; if, in addition, the puzzle's words make sense, it is said to be an ***alphametic***. The most well-known alphametic was published by the renowned British puzzlist Henry E. Dudeney (1857-1930):

$$\begin{array}{r} \mathrm{S\,E\,N\,D} \\ +\,\mathrm{M\,O\,R\,E} \\ \hline \mathrm{M\,O\,N\,E\,Y} \end{array}$$

    Two conditions are assumed: First, the correspondence between letters and digits is one-to-one, that is each letter represents one digit only and different letters represent different digits. Second, the digit zero does not appear as the left-most digit in any of the numbers. To solve an alphametic means to find which digit each letter represents. Note that a solution's uniqueness cannot be assumed and has to be verified by the solver.

    a. Write a program for solving cryptarithms by exhaustive search. Assume that a given cryptarithm is a sum of two words.

    b. Solve Dudeney'e puzzle the way it was expected to be solved when it was first published in 1924.

Hints to Exercises 3.4

1. a. Identify the algorithm's basic operation and count the number of times it will be executed.

   b. For each of the time amounts given, find the largest value of n for which this limit will not be exceeded.

2. How different is the traveling salesman problem from the problem of finding a Hamiltonian circuit?

3. Your algorithm should check the well-known conditions that are both necessary and sufficient for the existence of a Eulerian circuit in a connected graph.

4. Generate the remaining $4! - 6 = 18$ possible assignments, compute their costs, and find the one with the smallest cost.

5. Make the size of your counterexample as small as possible.

6. Rephrase the problem so that the sum of elements in one subset, rather than two, needs to be checked on each try of a possible partition.

7. Follow the definitions of a clique and of an exhaustive-search algorithm.

8. Try all possible orderings of the elements given.

9. Use common formulas of elementary combinatorics.

10. a. Add all the elements in the magic square in two different ways.

    b. What combinatorial objects do you have to generate here?

11. a. For testing, you may use alphametic collections available on the Internet.

    b. Given the absence of electronic computers in 1924, you must refrain here from using the Internet.