Exercises

1. Do some research on al-Khorezmi (also al-Khwarizmi), the man from whose name the word "algorithm" is derived. In particular, you should learn what the origins of the words "algorithm" and "algebra" have in common.

2. Given that the official purpose of the U.S. patent system is the promotion of the "useful arts," do you think algorithms are patentable in this country? Should they be?

3. **a.** Write down driving directions for going from your school to your home with the precision required from an algorithm's description.

   b. Write down a recipe for cooking your favorite dish with the precision required by an algorithm.

4. Design an algorithm for computing $\lfloor \sqrt{n} \rfloor$ for any positive integer n. Besides assignment and comparison, your algorithm may only use the four basic arithmetical operations.

5. Design an algorithm to find all the common elements in two sorted lists of numbers. For example, for the lists 2, 5, 5, 5 and 2, 2, 3, 5, 5, 7, the output should be 2, 5, 5. What is the maximum number of comparisons your algorithm makes if the lengths of the two given lists are m and n, respectively?

6. **a.** Find gcd(31415, 14142) by applying Euclid's algorithm.

b. Estimate how many times faster it will be to find gcd(31415, 14142) by Euclid's algorithm compared with the algorithm based on checking consecutive integers from $\min\{m, n\}$ down to $\gcd(m, n)$.

7. Prove the equality $\gcd(m, n) = \gcd(n, m \bmod n)$ for every pair of positive integers m and n.

8. What does Euclid's algorithm do for a pair of integers in which the first is smaller than the second? What is the maximum number of times this can happen during the algorithm's execution on such an input?

9. **a.** What is the minimum number of divisions made by Euclid's algorithm among all inputs $1 \le m, n \le 10$?

b. What is the maximum number of divisions made by Euclid's algorithm among all inputs $1 \le m, n \le 10$?

10. **a.** Euclid's algorithm, as presented in Euclid's treatise, uses subtractions rather than integer divisions. Write pseudocode for this version of Euclid's algorithm.

b. *Euclid's game* (see [Bog]) starts with two unequal positive integers on the board. Two players move in turn. On each move, a player has to write on the board a positive number equal to the difference of two numbers already on the board; this number must be new, i.e., different from all the numbers already on the board. The player who cannot move loses the game. Should you choose to move first or second in this game?

11. The ***extended Euclid's algorithm*** determines not only the greatest common divisor d of two positive integers m and n but also integers (not necessarily positive) x and y, such that $mx + ny = d$.

a. Look up a description of the extended Euclid's algorithm (see, e.g., [KnuI, p. 13]) and implement it in the language of your choice.

b. Modify your program to find integer solutions to the Diophantine equation $ax + by = c$ with any set of integer coefficients a, b, and c.

12. *Locker doors* There are n lockers in a hallway, numbered sequentially from 1 to n. Initially, all the locker doors are closed. You make n passes by the lockers, each time starting with locker #1. On the ith pass, $i = 1, 2, \ldots, n$, you toggle the door of every ith locker: if the door is closed, you open it; if it is open, you close it. After the last pass, which locker doors are open and which are closed? How many of them are open?

Exercises

1. *Old World puzzle* A peasant finds himself on a riverbank with a wolf, a goat, and a head of cabbage. He needs to transport all three to the other side of the river in his boat. However, the boat has room for only the peasant himself and one other item (either the wolf, the goat, or the cabbage). In his absence, the wolf would eat the goat, and the goat would eat the cabbage. Solve this problem for the peasant or prove it has no solution. (Note: The peasant is a vegetarian but does not like cabbage and hence can eat neither the goat nor the cabbage to help him solve the problem. And it goes without saying that the wolf is a protected species.)

2. *New World puzzle* There are four people who want to cross a rickety bridge; they all begin on the same side. You have 17 minutes to get them all across to the other side. It is night, and they have one flashlight. A maximum of two people can cross the bridge at one time. Any party that crosses, either one or two people, must have the flashlight with them. The flashlight must be walked back and forth; it cannot be thrown, for example. Person 1 takes 1 minute to cross the bridge, person 2 takes 2 minutes, person 3 takes 5 minutes, and person 4 takes 10 minutes. A pair must walk together at the rate of the slower person's pace. (Note: According to a rumor on the Internet, interviewers at a well-known software company located near Seattle have given this problem to interviewees.)

3. Which of the following formulas can be considered an algorithm for computing the area of a triangle whose side lengths are given positive numbers a, b, and c?

   a. $S = \sqrt{p(p-a)(p-b)(p-c)}$, where $p = (a+b+c)/2$

   b. $S = \frac{1}{2}bc \sin A$, where A is the angle between sides b and c

   c. $S = \frac{1}{2}ah_a$, where h_a is the height to base a

4. Write pseudocode for an algorithm for finding real roots of equation $ax^2 + bx + c = 0$ for arbitrary real coefficients a, b, and c. (You may assume the availability of the square root function $sqrt(x)$.)

5. Describe the standard algorithm for finding the binary representation of a positive decimal integer

   a. in English.

   b. in pseudocode.

6. Describe the algorithm used by your favorite ATM machine in dispensing cash. (You may give your description in either English or pseudocode, whichever you find more convenient.)

7. **a.** Can the problem of computing the number π be solved exactly?

   b. How many instances does this problem have?

   c. Look up an algorithm for this problem on the Internet.

8. Give an example of a problem other than computing the greatest common divisor for which you know more than one algorithm. Which of them is simpler? Which is more efficient?

9. Consider the following algorithm for finding the distance between the two closest elements in an array of numbers.

ALGORITHM *MinDistance*($A[0..n-1]$)
    //Input: Array $A[0..n-1]$ of numbers
    //Output: Minimum distance between two of its elements
    $dmin \leftarrow \infty$
        for $i \leftarrow 0$ **to** $n-1$ **do**
            for $j \leftarrow 0$ **to** $n-1$ **do**
                if $i \neq j$ **and** $|A[i] - A[j]| < dmin$
                $dmin \leftarrow |A[i] - A[j]|$
    return $dmin$

Make as many improvements as you can in this algorithmic solution to the problem. If you need to, you may change the algorithm altogether; if not, improve the implementation given.

10. One of the most influential books on problem solving, titled *How To Solve It* [Pol57], was written by the Hungarian-American mathematician George Pólya (1887–1985). Pólya summarized his ideas in a four-point summary. Find this summary on the Internet or, better yet, in his book, and compare it with the plan outlined in Section 1.2. What do they have in common? How are they different?

Important Problem Types

In the limitless sea of problems one encounters in computing, there are a few areas that have attracted particular attention from researchers. By and large, their interest has been driven either by the problem's practical importance or by some specific characteristics making the problem an interesting research subject.

The most important problem types are:

- Sorting
- Searching
- String processing
- Graph problems
- Combinatorial problems
- Geometric problems
- Numerical problems

Exercises

1. Consider the algorithm for the sorting problem that sorts an array by counting, for each of its elements, the number of smaller elements and then uses this information to put the element in its appropriate position in the sorted array:

ALGORITHM *ComparisonCountingSort*($A[0..n-1]$)

```
//Sorts an array by comparison counting
//Input: Array A[0..n − 1] of orderable values
//Output: Array S[0..n − 1] of A's elements sorted
//     in nondecreasing order
for i ← 0 to n − 1 do
    Count[i] ← 0
for i ← 0 to n − 2 do
    for j ← i + 1 to n − 1 do
        if A[i] < A[j]
            Count[j] ← Count[j] + 1
        else Count[i] ← Count[i] + 1
for i ← 0 to n − 1 do
    S[Count[i]] ← A[i]
return S
```

   a. Apply this algorithm to sorting the list 60, 35, 81, 98, 14, 47.

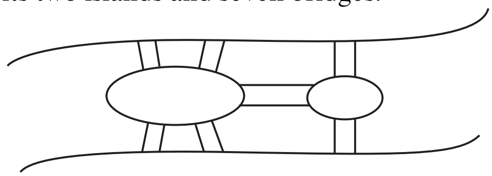
   b. Is this algorithm stable?

   c. Is it in-place?

2. Name the algorithms for the searching problem that you already know. Give a good succinct description of each algorithm in English. If you know no such algorithms, use this opportunity to design one.

3. Design a simple algorithm for the string-matching problem.

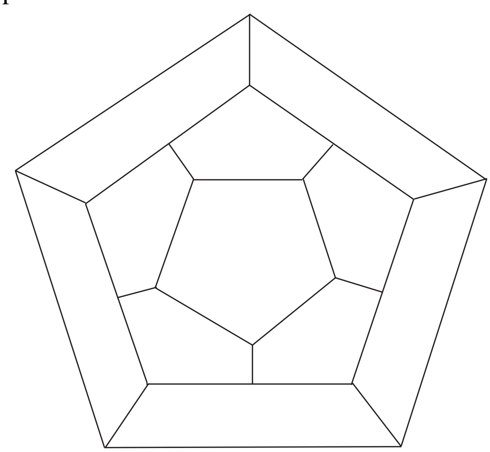
4. *Königsberg bridges* The Königsberg bridge puzzle is universally accepted as the problem that gave birth to graph theory. It was solved by the great Swiss-born mathematician Leonhard Euler (1707–1783). The problem asked whether one could, in a single stroll, cross all seven bridges of the city of Königsberg exactly once and return to a starting point. Following is a sketch of the river with its two islands and seven bridges:



a. State the problem as a graph problem.

b. Does this problem have a solution? If you believe it does, draw such a stroll; if you believe it does not, explain why and indicate the smallest number of new bridges that would be required to make such a stroll possible.

5. *Icosian Game* A century after Euler's discovery (see Problem 4), another famous puzzle—this one invented by the renowned Irish mathematician Sir William Hamilton (1805–1865)—was presented to the world under the name of the Icosian Game. The game's board was a circular wooden board on which the following graph was carved:



Find a ***Hamiltonian circuit***—a path that visits all the graph's vertices exactly once before returning to the starting vertex—for this graph.

6. Consider the following problem: Design an algorithm to determine the best route for a subway passenger to take from one designated station to another in a well-developed subway system similar to those in such cities as Washington, D.C., and London, UK.

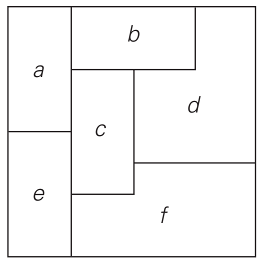
a. The problem's statement is somewhat vague, which is typical of real-life problems. In particular, what reasonable criterion can be used for defining the "best" route?

b. How would you model this problem by a graph?

7. a. Rephrase the traveling-salesman problem in combinatorial object terms.

b. Rephrase the graph-coloring problem in combinatorial object terms.

8. Consider the following map:



a. Explain how we can use the graph-coloring problem to color the map so that no two neighboring regions are colored the same.

b. Use your answer to part (a) to color the map with the smallest number of colors.

9. Design an algorithm for the following problem: Given a set of n points in the Cartesian plane, determine whether all of them lie on the same circumference.

10. Write a program that reads as its inputs the (x, y) coordinates of the endpoints of two line segments P_1Q_1 and P_2Q_2 and determines whether the segments have a common point.

Exercises

1. Describe how one can implement each of the following operations on an array so that the time it takes does not depend on the array's size n.
   a. Delete the ith element of an array ($1 \leq i \leq n$).
   b. Delete the ith element of a sorted array (the remaining array has to stay sorted, of course).
2. If you have to solve the searching problem for a list of n numbers, how can you take advantage of the fact that the list is known to be sorted? Give separate answers for
   a. lists represented as arrays.
   b. lists represented as linked lists.
3. **a.** Show the stack after each operation of the following sequence that starts with the empty stack:

   $$push(a), push(b), pop, push(c), push(d), pop$$

   b. Show the queue after each operation of the following sequence that starts with the empty queue:

   $$enqueue(a), enqueue(b), dequeue, enqueue(c), enqueue(d), dequeue$$

4. **a.** Let A be the adjacency matrix of an undirected graph. Explain what property of the matrix indicates that
   i. the graph is complete.
   ii. the graph has a loop, i.e., an edge connecting a vertex to itself.
   iii. the graph has an isolated vertex, i.e., a vertex with no edges incident to it.
   b. Answer the same questions for the adjacency list representation.
5. Give a detailed description of an algorithm for transforming a free tree into a tree rooted at a given vertex of the free tree.

6. Prove the inequalities that bracket the height of a binary tree with n vertices:

$$\lfloor \log_2 n \rfloor \le h \le n - 1.$$

7. Indicate how the ADT priority queue can be implemented as

a. an (unsorted) array.

b. a sorted array.

c. a binary search tree.

8. How would you implement a dictionary of a reasonably small size n if you knew that all its elements are distinct (e.g., names of the 50 states of the United States)? Specify an implementation of each dictionary operation.

9. For each of the following applications, indicate the most appropriate data structure:

a. answering telephone calls in the order of their known priorities

b. sending backlog orders to customers in the order they have been received

c. implementing a calculator for computing simple arithmetical expressions

10. *Anagram checking* Design an algorithm for checking whether two given words are anagrams, i.e., whether one word can be obtained by permuting the letters of the other. For example, the words *tea* and *eat* are anagrams.