HW 11 textbook problems and hints

Problem 1 6.5.11 [6.5.10] (6) (Factored form or not)

2 points for each part. Explain your answers.

11. According to the corollary of the Fundamental Theorem of Algebra, every polynomial

$$p(x) = a_n x^n + a_{n-1} x^{n-1} + \cdots + a_0$$

can be represented in the form

$$p(x) = a_n (x - x_1)(x - x_2) \cdots (x - x_n)$$

where $x_1, ..., x_n$ are the roots of the polynomial (generally, complex and not necessarily distinct). Discuss which of the two representations is more convenient for each of the following operations:

a. Polynomial evaluation at a given point

b. Addition of two polynomials

c. Multiplication of two polynomials

Author's hint:

Compare the number of operations needed to implement the task in question.

Problem 2 6.6.4a (6) (better than cubic check for length 3 cycle in graph)

4. a. Design an algorithm with a time efficiency better than cubic for checking whether a graph with n vertices contains a cycle of length 3 [Man89, p. 326].

Author's hint:

4. a. Base your algorithm on the following observation: a graph contains a cycle of length 3 if and only if it has two adjacent vertices i and j that are also connected by a path of length 2.

Problem 3 7.2.3 (6) (Horspool for binary strings)

3. How many character comparisons will be made by Horspool's algorithm in searching for each of the following patterns in the binary text of 1000 zeros?

   a. 00001

   b. 10000

   c. 01010

Author's hint:

3. For each pattern, fill in its shift table and then determine the number of character comparisons (both successful and unsuccessful) on each trial and the total number of trials.

Problem 4 7.2.7 (9) (Boyer-Moore for binary strings)

7. How many character comparisons will the Boyer-Moore algorithm make in searching for each of the following patterns in the binary text of 1000 zeros?

   a. 00001

   b. 10000

   c. 01010

Author's hint:

7. For each pattern, fill in the two shift tables and then determine the number of character comparisons (both successful and unsuccessful) on each trial and the total number of trials.

Problem 5: 7.2.8 (4) (does Boyer-Moore still work with just one table?)

8. a. Would the Boyer-Moore algorithm work correctly with just the bad-symbol table to guide pattern shifts?

   b. Would the Boyer-Moore algorithm work correctly with just the good-suffix table to guide pattern shifts?

Author's hint:

8. Check the description of the Boyer-Moore algorithm.

Problem 6: 7.2.11 (3, 5) (right cyclic shift)

11. You are given two strings S and T, each n characters long. You have to establish whether one of them is a right cyclic shift of the other. For example, PLEA is a right cyclic rotation of LEAP, and vice versa. (Formally, T is a right cyclic shift of S if T can be obtained by concatenating the $(n-i)$-character suffix of S and the i-character prefix of S for some $1 \leq i \leq n$.)

    a. Design a space-efficient algorithm for the task. Indicate the space and time efficiencies of your algorithm.

    b. Design a time-efficient algorithm for the task. Indicate the time and space efficiencies of your algorithm.

Author's hint:

11. a. A brute-force algorithm fits the bill here.

    b. Enhance the input before a search.

Problem 7: 7.3.4 (5) (probability that n keys all hash to the same table location)

4. Find the probability of all n keys being hashed to the same cell of a hash table of size m if the hash function distributes keys evenly among all the cells of the table.

Author's hint:

4. The question is quite similar to computing the probability of having the same result in n throws of a fair die.