BCB 5300

Homework 4: Pattern matching

1. As we discussed when going over assemblers, DNA sequencing reads contain errors that lead to complications in fragment assembly. Fragment assembly with errors motivates the Shortest k-Approximate Superstring Problem: given a set of strings S, find a shortest string s such that each string in S matches some substring of s with at most k errors. Design an approximation algorithm for this problem. (Hint: look back at some of our assembly graph-based methods again, like the Hamiltonian and Eulerian ones, and try to adapt them to this version of the problem.) What is the running time of your algorithm? Is your solution optimal, or is there an approximation guarantee if not?

2. A string $s = s_1 \ldots s_n$ is a *palindrome* if it spells the same string when read backwards, so that $s_i = s_{n+1-i}$ for all $1 \leq i \leq n$. Design an efficient algorithm to find all palindromes (of any length) in an input text T. What is the running time?

3. Use some suffix tree magic:

   (a) Design an algorithm to find the longest string shared by two given input texts T_1 and T_2. What is the running time of your algorithm?

   (b) Design an efficient algorithm to find the $s$hortest string in an input string T_1 that does not appear in a second input T_2. What is the running time?