Exercises for String Matching

Inge Li Gørtz

Bounds on the naive algorithm  Give an example that show that the worst case running time of the naive algorithm is $\Omega(nm)$. What is the best case running time of the naive algorithm?

Fingerprinting

- Show how to remove the assumption that the alphabet is binary. Assume the alphabet $\Sigma = \{1, 2, \ldots, u\}$. What if the alphabet is not integers?
- Show that $F(T_{s+1}) = 2 \cdot F(T_s) - 2^m T[s] + T[s + m + 1]$.

Cyclic rotations  Given two strings $P_1$ and $P_2$, determine if $P_1$ is a cyclic rotation of $P_2$. The string $P_1$ is a cyclic rotation of $P_2$ if $|P_1| = |P_2|$ and $P_1$ consists of a suffix of $P_2$ followed by a prefix of $P_2$. For example, arc is a cyclic rotation of car.

Circular strings  Give an algorithm to determine whether a string $P_1$ is a substring of a circular string $P_2$. A circular string of length $n$ is a string in which character $n$ is considered to precede character 1.

KMP preprocessing  Show we can find all occurrences of $P$ in $T$ by computing $\pi$ for $P$T, where $\$ is a symbol not in $\Sigma$.

Finite automaton preprocessing  Give an $O(m|\Sigma|)$ algorithm for computing the transition function $\delta$ for the string matching automaton for a pattern $P$.

Hint  Prove that $\delta(q, a) = \delta(\pi[q], a)$ if $q = m$ or $P[q + 1] \neq a$.

2-dimensional matching  Give an algorithm to solve the problem of looking for a $m \times m$ pattern $P$ in an $n \times n$ array of characters. The pattern can be shifted vertically and horizontally, but not rotated.

Linear time verification of Rabin-Karp  Let $L$ be the list of starting locations of probable occurrences of $P$ in $T$ returned by the Rabin-Karp algorithm. A run is a maximal interval of consecutive starting locations $l_1, l_2, \ldots, l_r$ from $L$ such that $l_{i+1} - l_i \leq m/2$. Consider a single run:

- A string $S$ is semiperiodic with period $\delta$ if $S$ consists of one or more copies of $\delta$ followed by a prefix of $\delta$. Show that if $P$ occurs at position $l_1$ and $l_2$ then $P$ is semiperiodic with a period of length $d = l_2 - l_1$.
- Show that $d$ is the smallest period of $P$.
- Show that if there are no false matches in the run, then $l_{i+1} - l_i = d$ for each $i$ in the run.
- Show that it suffices to check the previous condition plus explicitly checking each of the $d$ characters of $T$ starting at position $l_i + m - d$ against the last $d$ characters of $P$ for each $i$.  

Time analysis:

- Show that no character of $T$ is examined more than twice during a check of a single run.
- Show that no character of $T$ can be examined in the check of more than two runs.
- Conclude on the running time.

**Mandatory exercise**  Let $T$ be a tree, where each edge is labeled with one or more characters, and let $P$ be a string. Let $S$ be the set of subpaths of all root-to-leaf paths in $T$. Note that a subpath can start and end anywhere, also in the middle of an edge. The label of a subpath is the labels on the edges in the subpath. Give an efficient algorithm that finds all subpaths in $S$ that are labeled with pattern $P$. 