Advanced Data Structures

Labwork 5: Data structures for operations on strings

November 22, 2018

1. Construct the string-matching automaton for the pattern $P =$ `aabab` and illustrate its operation on the text string $T =$ `aaababaabaababaab`.

2. (Homework) Draw a state-transition diagram for a string-matching automaton for the pattern `ababbabbababbababbabb` over the alphabet $\{\texttt{a}, \texttt{b}\}$.

3. A **gap character** in a pattern P is a special character $\diamond$ that can match an arbitrary string of characters (even one of zero length). For example, the pattern $P = \texttt{ab}\diamond\texttt{ba}\diamond\texttt{c}$ occurs in the text `cabccbacbacab` as

$$\texttt{c}\underbrace{\texttt{ab}}_{\texttt{ab}}\underbrace{\texttt{cc}}_{\diamond}\underbrace{\texttt{ba}}_{\texttt{ba}}\underbrace{\texttt{cba}}_{\diamond}\underbrace{\texttt{c}}_{\texttt{c}}\texttt{ab} \quad \text{and as} \quad \texttt{c}\underbrace{\texttt{ab}}_{\texttt{ab}}\underbrace{\texttt{cc}}_{\diamond}\underbrace{\texttt{ba}}_{\texttt{ba}}\underbrace{\phantom{\texttt{ab}}}_{\diamond}\underbrace{\texttt{c}}_{\texttt{c}}\texttt{bacab}$$

   Given a pattern P containing gap characters, show how to build a finite automaton that can find an occurrence of P in a text T in $O(n)$ matching time, where $n = |T|$.

4. Construct the keyword tree and its failure links of the set of patterns

$$\mathcal{P} = \{\texttt{The}, \texttt{hand}, \texttt{and}, \texttt{pork}, \texttt{port}, \texttt{pot}\}.$$

   Indicate a string-matching automaton which recognizes the occurrences of patterns in $\mathcal{P}$.

5. (Homework) Construct the keyword tree and its failure links of the set of patterns $\mathcal{P} = \{\texttt{woman}, \texttt{man}, \texttt{meat}, \texttt{animal}\}$. Indicate a string-matching automaton which recognizes the occurrences of patterns in $\mathcal{P}$.

6. The construction of the transition function of the string matching automaton for $O[1..m]$ described in Lecture 7 has time complexity $O(m^3 \cdot |\Sigma|)$. There are better methods to construct the transition function, with time complexity $O(m \cdot |\Sigma|)$.

   Write down the pseudocode of an algorithm that constructs the transition function in time $O(m \cdot |\Sigma|)$, and prove that the complexity of your algorithm is $O(m \cdot |\Sigma|)$.

7. Draw the suffix tree and it suffix links for the text `banana$`.

8. (Homework) Draw the suffix tree and its suffix links for the text `mamaia$`.

9. (Homework) Draw the generalized suffix tree and its suffix links for the set of texts $\{\texttt{tatar}, \texttt{tabac}\}$.

Programming labwork

Write in C++ or Java a program which solves the following problem:

1. It reads a text T from a text file specified by the user
2. It reads from the terminal the number z of strings (patterns) $P_1, P_2, \ldots, P_z$
3. It reports all positions from T where there is an occurrence of a patterns P_i $(1 \leq i \leq z)$

The interaction of the user with the program should be as follows:

```
Enter the source file for the text: file-name
Enter the number of patterns: z
Enter pattern 1: P_1
...
Enter pattern z: P_z
```

Afterwards, the program displays the occurrences of every pattern in text the T which was read from the text file *file-name*:

```
Pattern 1 occurs at positions p_{1,1} ... p_{1,n_1}
...
Pattern z occurs at positions p_{z,1} ... p_{z,n_z}
```

The program should implement the Aho-Corasick algorithm which builds the keyword tree of the set of templates $\mathcal{P} = \{P_1, P_2, \ldots, P_z\}$ together with its failure links.

Illustrated example

Suppose that the filel `source.txt` contains the text

```
Tim a mers la Timisoara sa-si cumpere o casa.
```

If we specify

```
Enter the source file for the text: source.txt
Enter the number of patterns: 4
Enter pattern 1: Tim
Enter pattern 2: Timis
Enter pattern 3: sa
Enter pattern 4: casa
```

then the program must display

```
Pattern 1 occurs at positions 1 15
Pattern 2 occurs at positions 15
Pattern 3 occurs at positions 25 43
Pattern 4 occurs at positions 41
```