Lecture

At the lecture we will talk about string matching algorithms: the string matching automaton and the Knuth-Morris-Pratt algorithm (KMP). You should read CLRS section 32.0, 32.3, 32.4 (on Campusnet).

Exercises

1 Finite automata Construct both the string-matching automaton and the KMP automaton for the pattern $P = aabab$ and illustrate its operation on the text string $T = aaababaabaababaab$.
For KMP also write down the π-array.

2 KMP Solve

2.1 Compute the prefix function π for the pattern $ababbabbabbababbabb$ when the alphabet is $\Sigma = \{a, b\}$.and draw the corresponding automaton with failure links.

2.2 Explain how to determine the occurrences of pattern P in the text T by examining the π function for the string $P\$T$, where \$ is a new character not in the alphabet.

3 String matching with two strings Given two patterns P and P', describe how to construct a finite automaton that determines all occurrences of either pattern. Try to minimize the number of states in your automaton (CLRS 32.3-4.)

4 String matching with gaps In *string matching with gaps* the pattern P can contain a *gap character* $\star$ that can match *any* string (of arbitrary length even length zero). An example of such a string is $P = ab\star ac\star a$, which occurs in the text $T = bababacbcca$ in two ways:

T:	b	ab	ab	ac	bcc	a
P:		ab	$\star$	ac	$\star$	a

or

T:	bab	ab		ac	bcc	a
P:		ab	$\star$	ac	$\star$	a

There are no gap characters in the text—only in the pattern. Solve the following exercises.

4.1 Show how to build a finite automaton that can find an occurrence of a gapped pattern in P in a text T in $O(n)$ matching time.

4.2 Give an algorithm to find an occurrence of a pattern P containing gap characters in a text T in time $O(n+m)$. That is, preprocessing time + matching time should be $O(n+m)$).

5 Christmas songs (exam 2015) You are putting together a set of Christmas songs that will be handed out at the Christmas party. The Dean has declared that every song must contain the sentence "Merry␣Christmas␣Dear␣Dean", where "␣" denotes a blank space. E.g. the song:

```
We␣wish␣you␣a␣Merry␣Christmas␣
We␣wish␣you␣a␣Merry␣Christmas␣
We␣wish␣you␣a␣Merry␣Christmas␣
Dear␣Dean␣
Dear␣Dean
```

contains one occurrence of of the sentence "Merry␣Christmas␣Dear␣Dean" (line breaks are disregarded).

Formally, you are given a set S of songs $S_1, \ldots, S_k$ and a sentence P. Song S_i contains n_i characters and P contains m characters. Let $n = \sum_{i=1}^{k} n_i$ denote the total number of characters in the songs. All the strings are over an alphabet of size $O(1)$. Describe an algorithm that returns all the songs that contain P. Analyze the asymptotic running time of your algorithm. Remember to argue that your algorithm is correct.

6 Preprocessing of the string matching automaton Give an efficient algorithm for computing the transition function δ for the string-matching automaton corresponding to a given pattern P. Your algorithm should run in time $O(m|\Sigma|)$. (Hint: Prove that $\delta(q, a) = \delta(\pi[q], a)$ if $q = m$ or $P[q+1] \neq a$.)