Finite Representations of Languages

Lecture notes based on " Elements of the theory of computation" by H.R. Lewis and C. H. Papadimitriou.

The goal is to represent the languages with finite specifications. Consider an alphabet Σ, the number of finite representations of languages (it should be a string), and the number of languages over Σ.
No matter how powerful our methods for representing the languages, only countably many languages can be represented.

Definition 1 (Regular expressions) *The regular expressions over an alphabet Σ are all strings over the alphabet $\Sigma \cup \{(,), \emptyset, U, \star\}$ that can be obtained as follows:*
1. *$\emptyset$ and each member of Σ is a regular expression*
2. *If α and β are regular expressions than so is $(\alpha\beta)$*
3. *If α and β are regular expressions than so is $(\alpha U \beta)$*
4. *If α is a regular expression than so is $\alpha^\star$*
5. *Nothing is a regular expression unless it follows from 1-4*

Every regular expression defines a language. Symbols U and $\star$ are interpreted as union and Kleene star, respectively, and juxtaposition is string concatenation.

Definition 2 (Languages defined by regular expressions) *For a regular expression α, $\mathcal{L}(\alpha)$ is the language represented by α and it is defined as*
1. *$\mathcal{L}(\emptyset) = \emptyset$ and $\mathcal{L}(a) = \{a\}$ for each $a \in \Sigma$*
2. *If α and β are regular expressions than $\mathcal{L}(\alpha\beta) = \mathcal{L}(\alpha)\mathcal{L}(\beta)$*
3. *If α and β are regular expressions than $\mathcal{L}(\alpha U \beta) = \mathcal{L}(\alpha) \cup \mathcal{L}(\beta)$*
4. *If α is a regular expression than so is $\mathcal{L}(\alpha^\star) = \mathcal{L}(\alpha)^\star$*

Note that, even though $\alpha U \beta U \gamma$ and $\alpha\beta\gamma$ are not officially a regular expression (why?), we use it due to the associativity of concatenation and union operations.
The class of **regular languages** consists of all languages L such that $L = \mathcal{L}(\alpha)$ for some regular expression α. The class of regular languages over Σ is precisely the closure of the set of languages $\{\{\sigma\} \mid \sigma \in \Sigma\} \cup \{\emptyset\}$ with respect to union, concatenation, and Kleene star.

For some language L, an algorithm that answers the question is $w \in L$ is called a **language recognition device.** On the other hand, descriptions of how a string from a language can be produced are called **language generators**.

How can you prove that L is regular? How can you prove that L is not regular (later in the course).

See questions 1.8.1-1.8.5 (and the rest).

Related concepts: Closure

Definition 3 *Let D be a set, let $n \geq 0$, and let $R \subseteq D^{n+1}$ be a $(n+1)$-ary relation on D. Then a subset B of D is said to be* ***closed under*** *R if $b_{n+1} \in B$ whenever*
- *$b_1, \ldots, b_n \in B$ and*
- *$(b_1, \ldots, b_n, b_{n+1}) \in R$*

Any property of the form "the set B is closed under relations $R_1, \ldots, R_m$" is called a closure property of B.

Definition 4 *The closure of a relation R with respect to property P is the relation obtained by adding the minimum number of ordered pairs to R to obtain property P.*