

Strings

Notation: For every $n \in \mathcal{N}_+$ (the positive natural numbers), we will write $[n]$ to denote the set $\{1, 2, \dots, n\}$. Further, $[0]$ denotes the empty set, \emptyset .

Definitions: Given any (possibly infinite) set A , a *string over A* (that is, a string made up of symbols from A) is any finite sequence $u : [n] \rightarrow A$.

The set A is called the *alphabet*, and the natural number n is called the *length* of u , denoted $|u|$.

If $n > 0$ then for every $i \in [n]$ $u(i)$ is some element of A , also denoted u_i , and the string u is also denoted $u_1 \dots u_n$.

For $n = 0$, we have the string corresponding to the unique (empty) function from \emptyset to A , called *the null string* (or *the empty string*), denoted by ϵ_A .

Strings

Definitions: Given two strings $u : [m] \rightarrow A$ and $v : [n] \rightarrow A$, with $m, n \geq 0$, their *concatenation*, denoted $u.v$ or just uv , is the string $w : [m + n] \rightarrow A$ such that:

$$w(i) = \begin{cases} u(i) & \text{if } 1 \leq i \leq m \\ v(i - m) & \text{if } m + 1 \leq i \leq m + n \end{cases}$$

Clearly, ϵ is a left and right identity of the concatenation operation. That is, for every string u , $u = \epsilon.u = u.\epsilon$.

Given strings u and v , v is a *prefix* of u if there is a string w such that $u = vw$. It is a *suffix* of u if there is a string w such that $u = wv$. It is a *substring* of u if there are strings w and z such that $u = wvz$. A prefix/suffix/substring is *proper* if $v \neq u$ and $v \neq \epsilon$.

The set of all strings over a set A is denoted by A^* . If the empty string is excluded we have the set A^+ . These operators are called *Kleene Star* and *Kleene plus* after the logician Steven Kleene.

Trees

Definition: A *tree domain*, D , is a nonempty subset of strings in \mathcal{N}_+^* satisfying the following conditions:

1. For each $u \in D$, every prefix of u is also in D
2. For each $u \in D$, for every $i \in \mathcal{N}_+$, if $ui \in D$ then for every j , $1 \leq j < i$, uj is also in D .

Example: The tree domain

$$D = \{\epsilon, 1, 2, 11, 21, 22, 23, 221, 222, 2211, 231, 232\}$$

corresponds to the structure:

Trees

Definition: Given a set Σ of *labels*, a Σ -*tree* (or just *tree*) is a total function $t : D \rightarrow \Sigma$ (that is, a D -indexed sequence of elements of Σ), where D is a tree domain.

The domain of a tree t is denoted $dom(t)$. Each string in $dom(t)$ is called a *tree address* or a *node*. If $d \in dom(t)$, then the node d is labelled by the element $t(d)$ of Σ .

Example: Let $\Sigma = \{f, g, h, a, b\}$. Then the tree $t : D \rightarrow \Sigma$, where D is the tree domain from the last example, and t is the function whose graph is:

$$\begin{aligned} &\{(\epsilon, f), (1, h), (2, g), (1\ 1, a), (2\ 1, a), (2\ 2, f), (2\ 3, b), \\ &\quad (2\ 2\ 1, h), (2\ 2\ 2, b), (2\ 2\ 1\ 1, a), (2\ 3\ 1, b), (2\ 3\ 2, f)\} \end{aligned}$$

corresponds to the structure:

Trees

Definitions: The *outdegree* of a node u , denoted $d(u)$ is the cardinality of the set $\{i \mid ui \in \text{dom}(t)\}$, and can be infinite. A *leaf* is a node with outdegree 0. Thus the address of a leaf, u , is such that u is a proper prefix of no string in $\text{dom}(t)$.

The *root* of a tree is the node with address ϵ .

For any non-leaf node, u , each node of the tree t with an address in $\text{dom}(t)$ of the form ui (where $i \in \mathcal{N}_+$) is a *child* or *immediate successor* of the node u . Any address of which u is a proper prefix is a *descendant* of u , and u is an *ancestor* of that node, and u is said to *dominate* that node.

Two nodes with addresses of the form ui and uj , with $i, j \in \mathcal{N}_+$ and $i < j$, are said to be *siblings*. The node ui is the *left sibling* of uj and uj is the *right sibling* of ui .

Trees

Clearly, address prefix, or ancestorship, forms a partial order on the nodes of a tree.

However, Tree addresses can be totally ordered lexicographically as follows: $u < v$ if $v = uw$ for some non-empty string w (that is, if u is a prefix of v , and, therefore, the node u is an ancestor of the node v), or if there exist strings $x, y, z \in \mathcal{N}_+^*$ and numbers $i, j \in \mathcal{N}_+$ with $i < j$ such that $u = xiy$ and $v = xjz$ (that is, if the node u is “to the left of” the node v).

Trees

Definitions: A *finite path* with *source* u and *target* v (or *from* u *to* v) is a finite sequence of nodes u_0, u_1, \dots, u_n such that $u = u_0$, $v = u_n$, and for all j , $1 \leq j \leq n$, $u_j = u_{j-1}i_j$ for some $i_j \in \mathcal{N}_+$. The *length of the path* is n . A *branch* of the tree is a path from the root to a leaf.

An *infinite path* with source u (or *from* u), is an infinite sequence of nodes u_0, u_1, \dots such that $u = u_0$, and for all j , $1 \leq j$, $u_j = u_{j-1}i_j$ for some $i_j \in \mathcal{N}_+$.

Given a finite tree (a tree with finite domain), the *height* of the node u in $\text{dom}(t)$ is equal to

$$\max(\{\text{length}(p) \mid p \text{ is a path from } u \text{ to a leaf}\})$$

or, alternately:

The *depth* of a tree is the height of its root (the length of the longest path in the tree).