# Semantic tree operations

Let us introduce the *tree tuple* where represents certain node - semantic particle or a subtree.

Here is a tree tuple factor which encodes uniquely the position of the node in the tree.

The following operations are defined for tree tuple factors:

We have a base of tree tuple factors which are defined as the digits greater than 0 of -nary number system such that . We define an operation `` denoting digit concatenation . Obviously,

for any pair

Note that the latter implies that

for any tuple where

Encoding a complete -ary tree of height with the algebraic notation above:

. Further we will assume that .

In general we have:

where

Obviously, we have at most distinct terms which represent nodes i.e. semantic values.

The expression for the tree also can be written as:

where and are the *node factors* given with . The node values are the values ordered in increasing order of . This order corresponds to *level order traversal* of the -ary tree. Note that with appropriately defined comparison operation `<` we can model different ways of traversing the -ary tree. For instance, if we define `<` as the comparison for the values of we will have ordering which corresponds to the *preorder traversal* of the tree.

*Example*

Peter is Dimitar’s son.

Dimitar’s son has a friend in the neighborhood and his friend’s name is James.

* James is Peter’s friend

Peter is the son of Dimitar.

The son of Dimitar has a friend in the neighborhood and the name of his friend is James.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

/ \ / \

\_\_\_\_\_\_\_\_

/ | \ / \ \ / \

/ | \ | / \ / \

/ \

T

V

/ \

/ | \

V V

Expressing T1 with the algebraic notation discussed earlier:

(k0,T1) = (k0,( (k0, v2) + (k1, v1) + (k2, ( (k0, v3) + (k1, v4)

Semantic Aggregation:

\_\_\_\_\_\_\_\_\_

/ \ / \

\_\_\_\_\_ V

/ \ \ / \

V V

Properties and Dependent Properties:

P

\ | /

P--- V --- P

/ | \

P P P