NAME

grammar::fa::dacceptor − Create and use deterministic acceptors

SYNOPSIS

package require **Tcl 8.4**

package require **snit**

package require **struct::set**

package require **grammar::fa::dacceptor ?0.1.1?**

::grammar::fa::dacceptor *daName fa* ?**-any** *any*?

daName *option* ?*arg arg ...*?

daName **destroy**

daName **accept?** *symbols*

DESCRIPTION

This package provides a class for acceptors constructed from deterministic *finite automatons* (DFA). Acceptors are objects which can be given a string of symbols and tell if the DFA they are constructed from would *accept* that string. For the actual creation of the DFAs the acceptors are based on we have the packages **grammar::fa** and **grammar::fa::op**.

API

The package exports the API described here.

::grammar::fa::dacceptor *daName fa* ?**-any** *any*?

Creates a new deterministic acceptor with an associated global Tcl command whose name is *daName*. This command may be used to invoke various operations on the acceptor. It has the following general form:

daName *option* ?*arg arg ...*?

Option and the *arg*s determine the exact behavior of the command. See section **ACCEPTOR METHODS** for more explanations.

The acceptor will be based on the deterministic finite automaton stored in the object *fa*. It will keep a copy of the relevant data of the FA in its own storage, in a form easy to use for its purposes. This also means that changes made to the *fa* after the construction of the acceptor *will not* influence the acceptor.

If *any* has been specified, then the acceptor will convert all symbols in the input which are unknown to the base FA to that symbol before proceeding with the processing.

ACCEPTOR METHODS

All acceptors provide the following methods for their manipulation:

daName **destroy**

Destroys the automaton, including its storage space and associated command.

daName **accept?** *symbols*

Takes the list of *symbols* and checks if the FA the acceptor is based on would accept it. The result is a boolean value. **True** is returned if the symbols are accepted, and **False** otherwise. Note that bogus symbols in the input are either translated to the *any* symbol (if specified), or cause the acceptance test to simply fail. No errors will be thrown. The method will process only just that prefix of the input which is enough to fully determine (non-)acceptance.

EXAMPLES

BUGS, IDEAS, FEEDBACK

This document, and the package it describes, will undoubtedly contain bugs and other problems. Please report such in the category *grammar_fa* of the *Tcllib Trackers* [http://core.tcl.tk/tcllib/reportlist]. Please also report any ideas for enhancements you may have for either package and/or documentation.

KEYWORDS

acceptance, acceptor, automaton, finite automaton, grammar, parsing, regular expression, regular grammar, regular languages, state, transducer

CATEGORY

Grammars and finite automata

COPYRIGHT