

Written by : Barak Weichselbaum, all rights reserved 2000-2008

Site: <http://www.komodia.com/>

Email : barak@komodia.com

Komodias implementation of Aho-Corasick algorithm

By Barak Weichselbaum

Introduction

This manual covers how to use the CSuffixTrie class which implements the Aho-Corasick search algorithm (http://en.wikipedia.org/wiki/Aho-Corasick_algorithm)

Overview

I implemented this class to facilitate quick keyword search for parental control - to be integrated with "Komodia's redirector" product (<http://www.komodia.com/index.php?page=redirector.html>)
The class is intended to be thread safe while read only, which is the normal usage for parental control after calculating the initial modified suffix trie.

Usage

AddString – Adding a search string to the trie.

FindString – Search if a string is inside the trie.

DeleteString – Delete a string from the trie.

BuildTreeIndex – Once done adding/removing strings this method is called to generate the trie used in the algorithm search, it's important to note that you must call this function manually; calling the previous three string manipulation routines will not cause the trie to be recalculated since this is a time intensive method.

Clear – Will clear all strings in the trie.

GetAllStringsVector – Get all the strings stores in the trie in a vector format.

GetAllStringsSet - Get all the strings stores in the trie in a set format.

SearchAhoCorasik – Perform an Aho-Corasick search on a given string; will break when first occurrence is found.

SearchAhoCorasikMultiple - Perform an Aho-Corasick search on a given string - will find all occurrences.