HELSINGIN YLIOPISTO
HELSINGFORS UNIVERSITET
UNIVERSITY OF HELSINKI
MATEMAATTIS-LUONNONTIETEELLINEN TIEDEKUNTA
MATEMATISK-NATURVETENSKAPLIGA FAKULTETEN
FACULTY OF SCIENCE

String Sorting in Python – Comparison of Several Algorithms

Onni Koskinen, Arturs Polis, and Lari Rasku

TESTING DATA

Dataset	Number of strings	Sum of lengths	Max string length	alphabet size	Sum of LCP array
dna.100MB	618	104856983	3732300	15	4501
dna.200MB	1114	209714087	3732300	15	8948
proteins.100MB	359505	104498096	36805	24	18853436
proteins.200MB	709116	209006085	36805	24	50076184
urls.100MB	3284368	101569109	372	114	94113004
urls.200MB	6576059	203139142	560	114	191545831
words.100MB	18502734	85200064	112	211	83643408
words.200MB	37003241	170395992	112	220	168115390

Table 1: Data set used for comparing the algorithms

TEST RESULTS

Dataset	Number of strings	Sum of lengths	Max string length	alphabet size	Sum of LCP array
dna.100MB	618	104856983	3732300	15	4501
dna.200MB	1114	209714087	3732300	15	8948
proteins.100MB	359505	104498096	36805	24	18853436
proteins.200MB	709116	209006085	36805	24	50076184
urls.100MB	3284368	101569109	372	114	94113004
urls.200MB	6576059	203139142	560	114	191545831
words.100MB	18502734	85200064	112	211	83643408
words.200MB	37003241	170395992	112	220	168115390

Table 2: Algorithm running times

PERFORMANCE GRAPHS

The graphs below show the time and space requirements of several algorithms on two texts. The algorithms are divided into three groups:

New algorithms based on reference point ranks, repetition shortcuts and wavelet trees

Improved implementations of wavelet trees and algorithms from [?]

Prior algorithms from [?, ?]