Pavel Samsonov

10 May 2017

CSS 342

Program 5 report

Driver.cpp output:

#faculty members: 10

contents:

-inf -inf -inf -inf -inf -inf

berger berger berger

cioch

erdly erdly erdly erdly erdly

fukuda

jackels

olson olson olson

stiber

sung

unknown unknown

zander zander

+inf +inf +inf +inf +inf +inf

deleting unknown

#faculty members: 9

contents:

-inf -inf -inf -inf -inf -inf

berger berger berger

cioch

erdly erdly erdly erdly erdly

fukuda

jackels

olson olson olson

stiber

sung

zander zander

+inf +inf +inf +inf +inf +inf

finding stiber = 1

create another list

finding stiber = 1

#faculty members: 9

cost of find = 121

Performance results:

dlist’s find cost = 6,491,439

mtflist’s find cost = 66,448

translist’s find cost = 6,422,436

skip’s find cost 10,982,613

Seeing as statistics randomly accesses list elements, it would make sense that skip has the highest cost because it does not randomly access elements. However, if we account for that theoretically skip should have the fastest access time for find. Find on a skiplist should have a access time of log n will. The best case for MTF is 1 and worst case is n. Transpose has the same best and worst case scenarios however its average is better than MTF list. Accounting for the average cases skiplist should have a find time of about log n. For average scenarios skiplist is better and for worst case skiplist is better. However, since stats can use random access for the other lists it will have a much faster access time than the skiplist.