

VILNIAUS UNIVERSITETAS
MATEMATIKOS IR INFORMATIKOS FAKULTETAS
INFORMATIKOS INSTITUTAS
INFORMATIKOS KATEDRA

Baigiamasis bakalauro darbas

Rikiavimo tobulinimas genetiniais algoritmais
(Improving Sorting with Genetic Algorithms)

Atliko: 4 kurso 2 grupės studentas

Deividas Zaleskis (parašas)

Darbo vadovas:

Irmantas Radavičius (parašas)

Recenzentas:

doc. dr. Vardauskas Pavardauskas (parašas)

Vilnius
2022

Turinys

Sąvokų apibrėžimai	2
Įvadas	3
1. Pagrindinė tiriamoji dalis	4
Išvados	5
Conclusions	6

Sąvokų apibrėžimai

Sutartinių ženklų, simbolių, vienetų ir terminų sutrumpinimų sąrašas (jeigu ženklų, simbolių, vienetų ir terminų bendras skaičius didesnis nei 10 ir kiekvienas iš jų tekste kartojasi daugiau nei 3 kartus).

Įvadas

Duomenų rikiavimas yra vienas aktyviausiai tiriamų uždavinių informatikos moksle. Iš dalies tai lemia rikiavimo uždavinio prieinamumas ir sąlygos paprastumas. Formaliai rikiavimo uždavinys formuluojamas taip: duotai baigtinei palyginamų elementų sekai $S = (s_1, s_2, \dots, s_n)$ pateikti tokį kėlinį, kad duotosios sekos elementai būtų išdėstyti monotoniškai (didėjančia arba mažėjančia) tvarka. Kadangi rikiavimo uždavinio sąlyga yra paprasta, tai suteikia didelę galimų implementacijų įvairovę. Todėl nauji rikiavimo algoritmai ir įvairūs patobulinimai egzistuojantiems algoritmams yra kuriami ir dabar.

Rikiavimo uždavinys yra fundamentalus, kadangi efektyvus rikiavimas padeda pagrindą efektyviam kitų uždavinių sprendimui. Kaip to pavyzdį galima pateikti dvejetainės paieškos algoritmą, kurio prielaida, jog duomenys yra išrikiuoti, leidžia sumažinti paieškos laiko sudėtingumą iki $O(\log n)$. Efektyvus rikiavimas taip pat svarbus duomenų normalizavimui bei pateikimui žmonėms lengvai suprantamu formatu. Kadangi duomenų rikiavimas yra fundamentalus uždavinys, net ir nežymūs patobulinimai žvelgiant bendrai gali atnešti didelę naudą.

Rikiavimo uždaviniui spręsti egzistuoja labai įvairių algoritmų. Plačiausiai žinomi yra klasikiniai rikiavimo algoritmai: rikiavimas sąlaja (angl. merge sort), rikiavimas įterpimu (angl. insertion sort) ir greitojo rikiavimo algoritmas (angl. quicksort). Tiesa, šie algoritmai turi įvairių trūkumų: rikiavimas sąlaja veikia lėčiau nei nestabilūs algoritmai, rikiavimas įterpimu yra efektyvus tik kai rikiuojamų duomenų dydis yra mažas, o greitojo rikiavimo algoritmas netinkamai parinkus slenksčių gali degraduoti į $O(n^2)$ laiko sudėtingumą. Todėl šiuo metu praktikoje plačiausiai naudojami yra hibridiniai rikiavimo algoritmai, kurie apjungia kelis klasikinius algoritmus į vieną panaudodami jų geriausias savybes. Pavyzdžiui, C++ programavimo kalbos standartinėje bibliotekoje pateikiamas introsort algoritmas įprastai naudoja greitojo rikiavimo algoritmą, pasiekus tam tikrą rekursijos gylį yra naudojamas rikiavimas krūva (angl. heapsort) siekiant išvengti degradacijos į $O(n^2)$ laiko sudėtingumą, o kai rikiuojamų duomenų dydis yra pakankamai mažas, pasitelkiamas rikiavimas įterpimu, kadangi su mažais duomenų dydžiais jis yra efektyvesnis. Apibendrinant, pasitelkiant įvairius rikiavimo algoritmus ir jų unikalias savybes yra įmanoma rikiavimo uždavinį spręsti efektyviau.

1. Pagrindinė tiriamoji dalis

Pagrindinėje tiriamojoje dalyje aptariama ir pagrindžiama tyrimo metodika; pagal atitinkamas darbo dalis, nuosekliai, panaudojant lyginamosios analizės, klasifikacijos, sisteminimo metodus bei apibendrinimus, dėstoma sukaupta ir išanalizuota medžiaga.

Išvados

Išvadose ir pasiūlymuose, nekartojant atskirų dalių apibendrinimų, suformuluojamos svarbiausios darbo išvados, rekomendacijos bei pasiūlymai.

Conclusions

Šiame skyriuje pateikiamos išvados (reziumė) anglų kalba.