

Paieškos Algoritmai

Šiandien panagrinėsime paieškos algoritmus. Aptarsime jų veikimo principus ir panaudojimo sritis. Išsiaiškinsime, kada kuris algoritmas yra efektyviausias. Pamoka skirta studentams, besidomintiems duomenų struktūromis ir algoritmais.

AG

by Andrej Gorbatniov



Kas yra paieška?

Paieška – tai procesas, skirtas rasti konkretų elementą duomenų rinkinyje. Tai gali būti failų paieška kompiuteryje arba įrašų duomenų bazėje.

Pavyzdžiui, paieška internete, failų sistemoje, duomenų bazėje. Paieškos algoritmai optimizuoja šį procesą.





Paieškos Algoritmų Skirstymas

Nuoseklios paieškos metodai

Peržiūrimi visi elementai paeiliui, kol randamas ieškomasis.

Tiesioginė prieiga (Hashing)

Elementai pasiekiami tiesiogiai pagal rakto reikšmę.

Paieška medžio struktūrose

Naudojamos medžio duomenų struktūros, pvz., dvejetainis paieškos medis.

Kiekvienas metodas turi savo privalumų ir trūkumų, priklausomai nuo duomenų struktūros ir dydžio.

Nuosekli Paieška

1

Apibrėžimas

Peržiūrimi visi elementai iš eilės.

2

Veikimo principas

Pradedama nuo pirmo elemento ir tęsiama iki galo, arba kol randamas ieškomasis.

3

Privalumai

Paprasta, tinka nestruktūruotiems duomenims.

4

Trūkumai

Lėta dideliems duomenų masyvams.

Šuoliuojanti Paieška

Apibrėžimas

Tai paieška, kuri peršoka blokus, kad surastų ieškomą elementą. Tai greitesnė už nuoseklią paiešką.

Kaip veikia?

Peršokama po blokus, kol randamas blokas, kuriame gali būti ieškomasis elementas.

Pavyzdys

Jei ieškome 55, peršokame blokus po 10 elementų, kol randame bloką, kuriame gali būti 55.



Dvejtainė Paieška

1

Veikimo principas

Duomenų masyvas dalijamas per pusę. Jei ieškomasis elementas didesnis už vidurinį, ieškoma dešinėje pusėje, priešingu atveju – kairėje. Veiksmai kartojami, kol elementas randamas.

2

Privalumai

Greita paieška surūšiuotuose duomenyse.

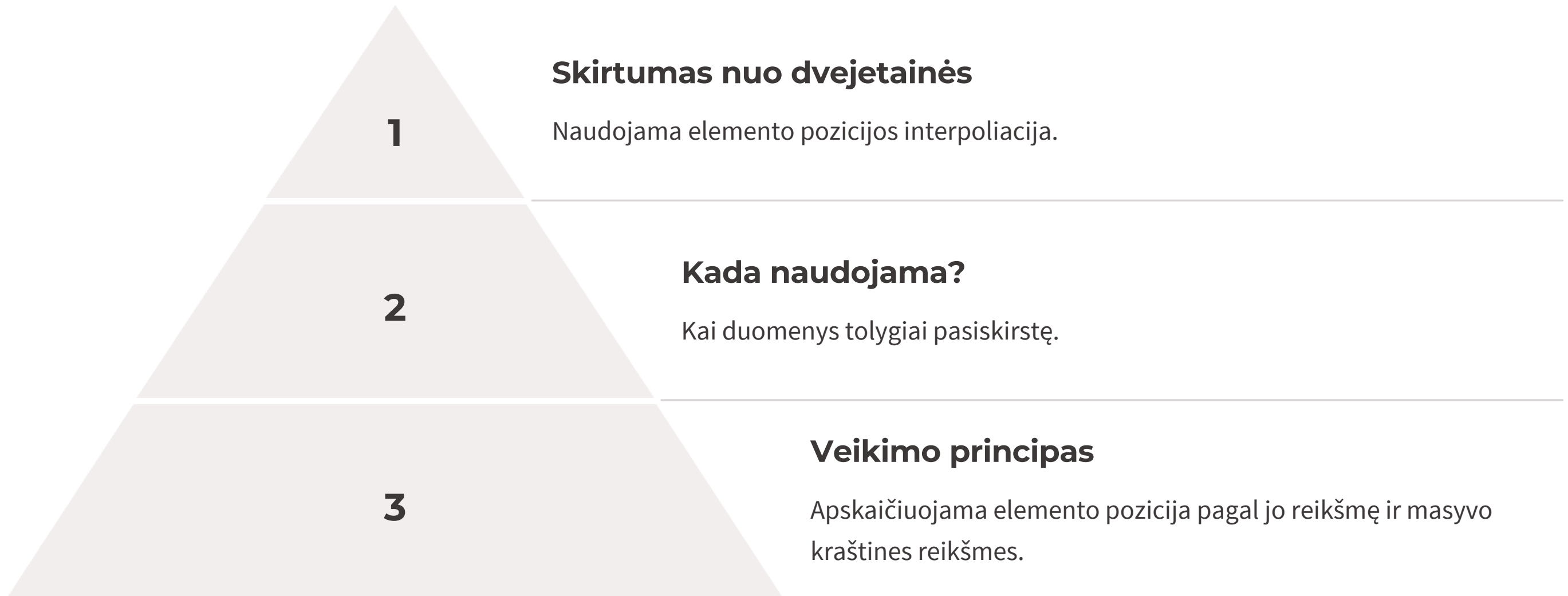
3

Trūkumai

Reikalauja surūšiuotų duomenų.



Interpoliacinė Paieška



Interpoliacinė paieška efektyvesnė už dvejetainę paiešką, kai duomenys tolygiai pasiskirstę, tačiau gali būti lėtesnė, kai duomenys netolygūs.

Algoritmų Palyginimas

Algoritmas	Sudėtingumas	Privalumai	Trūkumai
Nuosekli paieška	$O(n)$	Paprasta, tinka nestruktūruotiems duomenims	Lėta dideliems masyvams
Šuoliuojanti paieška	$O(\sqrt{n})$	Greitesnė nei nuosekli	Reikalauja surūšiavimo
Dvejtainė paieška	$O(\log n)$	Greita surūšiuotiems duomenims	Netinka nestruktūruotiems duomenims
Interpoliacinė paieška	$O(\log \log n)$	Greitesnė už dvejtainę kai duomenys tolygiai pasiskirstę	Netinka netolygiai pasiskirsčiusiems duomenims

Šioje lentelėje pateiktas algoritmų sudėtingumo, privalumų ir trūkumų palyginimas. Tai padeda pasirinkti tinkamiausią algoritmą pagal duomenų pobūdį ir dydį.



Apibendrinimas

$O(n)$

Nuosekli paieška

$O(\log n)$

Dvejetainė paieška

$O(\log \log n)$

Interpoliacinė paieška

Kada naudoti kokį algoritmą? Kuri paieška efektyviausia? Atsakymai priklauso nuo duomenų struktūros ir dydžio.

Klausimas: Ar visada verta rūšiuoti duomenis prieš paiešką?