



Paieškos Algoritmai: Nuosekli ir Dvejetainė Paieška

Šiandien gilinsimės į paieškos algoritmų pasaulį. Aptarsime nuoseklios ir dvejetainės paieškos principus. Išmoksime praktiškai pritaikyti abu algoritmus. Analizuosime jų efektyvumą skirtinguose scenarijuose. Pasiruoškite interaktyviai pamokai!

AG

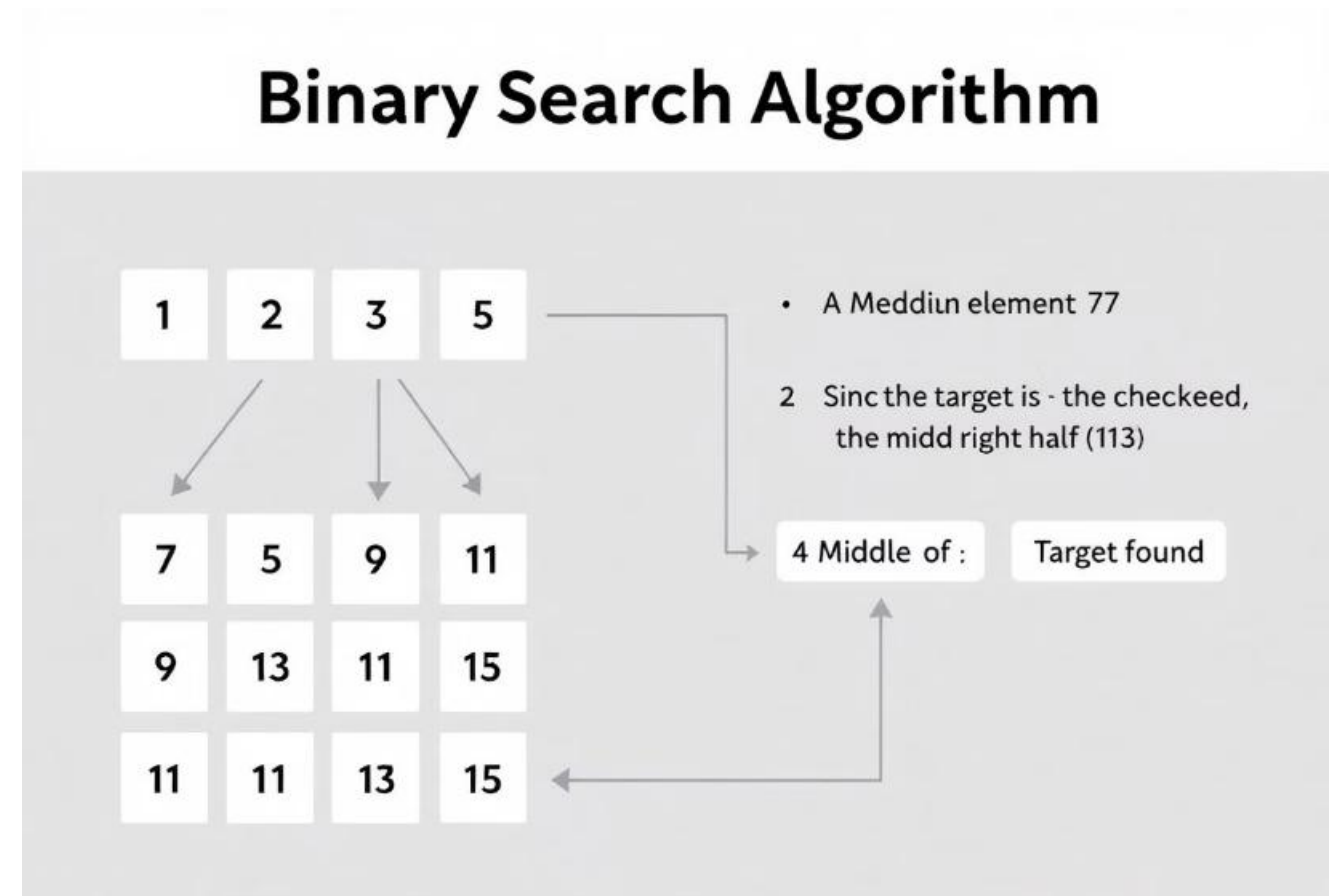
by Andrej Gorbatniov

Paieškos Algoritmų Svarba

Kas yra paieškos algoritmai?

Tai metodai, skirti rasti specifinę informaciją duomenų rinkinyje.

Jie veikia tikrindami duomenų elementus, kol suranda atitinkamą.



Kodėl jie svarbūs?

Efektyvūs algoritmai taupo laiką ir resursus. Paieška yra pagrindinė daugelio programų dalis.

Nuosekli Paieška: Principas ir Savybės

Veikimo principas

Tikrinama kiekviena elemento reikšmė iš eilės. Ieškoma, kol randamas tikslinis elementas arba sąrašas baigiasi.

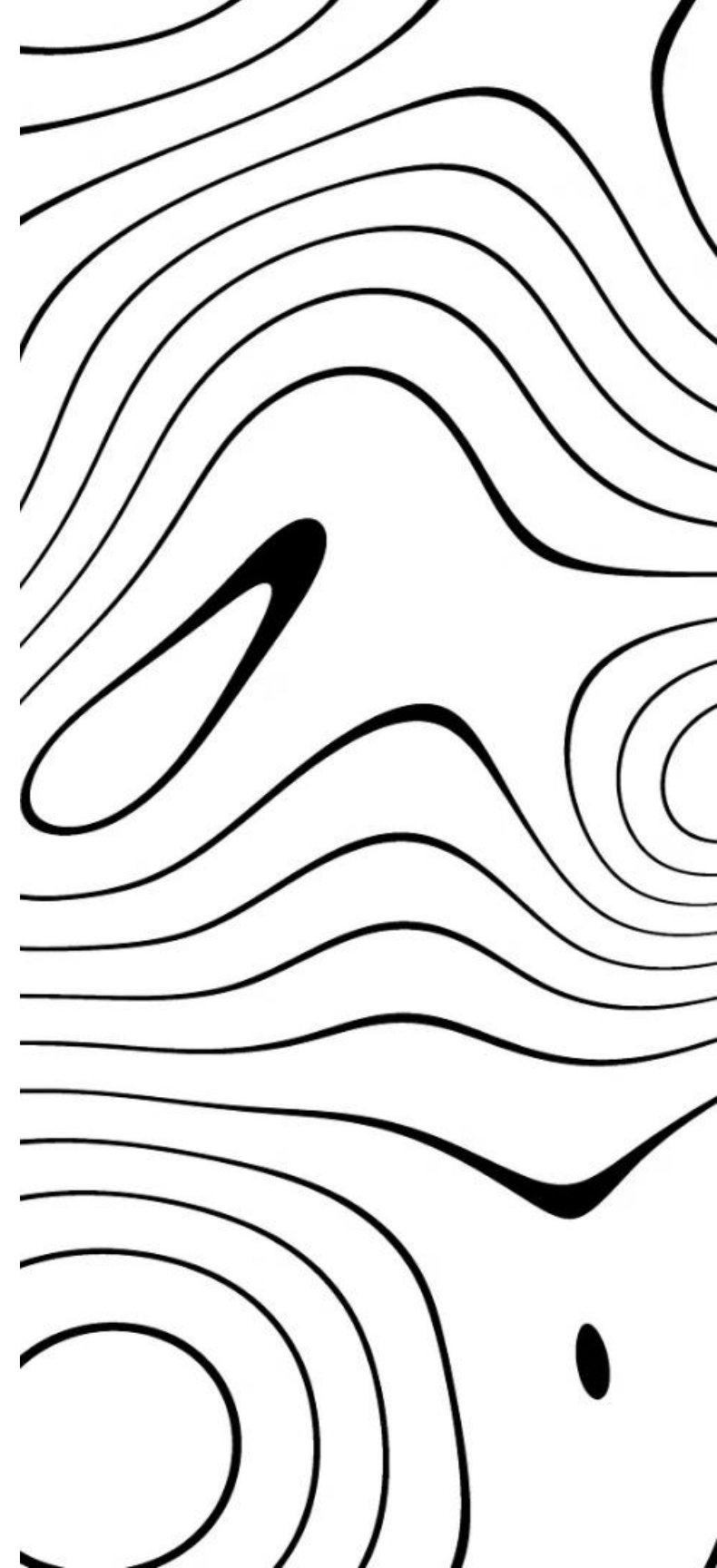
Privalumai

Veikia su bet kokia duomenų struktūra. Algoritmas paprastas suprasti ir įgyvendinti.

Trūkumai

Lėtas dideliuose duomenų rinkiniuose. Veikimo laikas yra $O(n)$.

Nuosekli paieška yra tiesioginis, bet ne visada efektyvus metodas.



Nuoseklios Paieškos Pseudo Kodas

```
FUNKCIJA NuosekliPaieska(masyvas, ieskomaReiksme):  
  UZ kiekviena elementa masyve:  
    JEI elementas == ieskomaReiksme:  
      GRAŽINTI indeksą  
    GRAŽINTI -1 (elementas nerastas)
```

Pseudo kodas aiškiai parodo algoritmo veiksmus. Jis leidžia lengvai suprasti logiką.

```
{  
1  if if//else if - if d//eilses control {  
    });  
2  if if-pentice nastaul({  
    iws uting struction (if));  
1  }>  
2  if_ifr/else control in {f);  
}  
><  
11  if rauisa if {;  
    oncentration_purinmotion.id);  
3  if ielss(f{  
    {raw"t,  
16    is auticattion, "d);  
    to deuatiln, 'if;  
12    if flow up,tine nate(),  
    }>  
15    ' ir_wial,ebutzh-('fit");  
    >>>  
    };  
11    "the-cond fi{f;  
    };  
    };  
12    if ifelsse(oducations:"t));  
    });  
11  }  
13  if realsse control,;  
    };  
15  }
```

Dvejtainė Paieška: Efektyvus Metodas

Veikimo principas

Dalijamas surūšiuotas masyvas per pusę. Tikrinama vidurinė reikšmė.

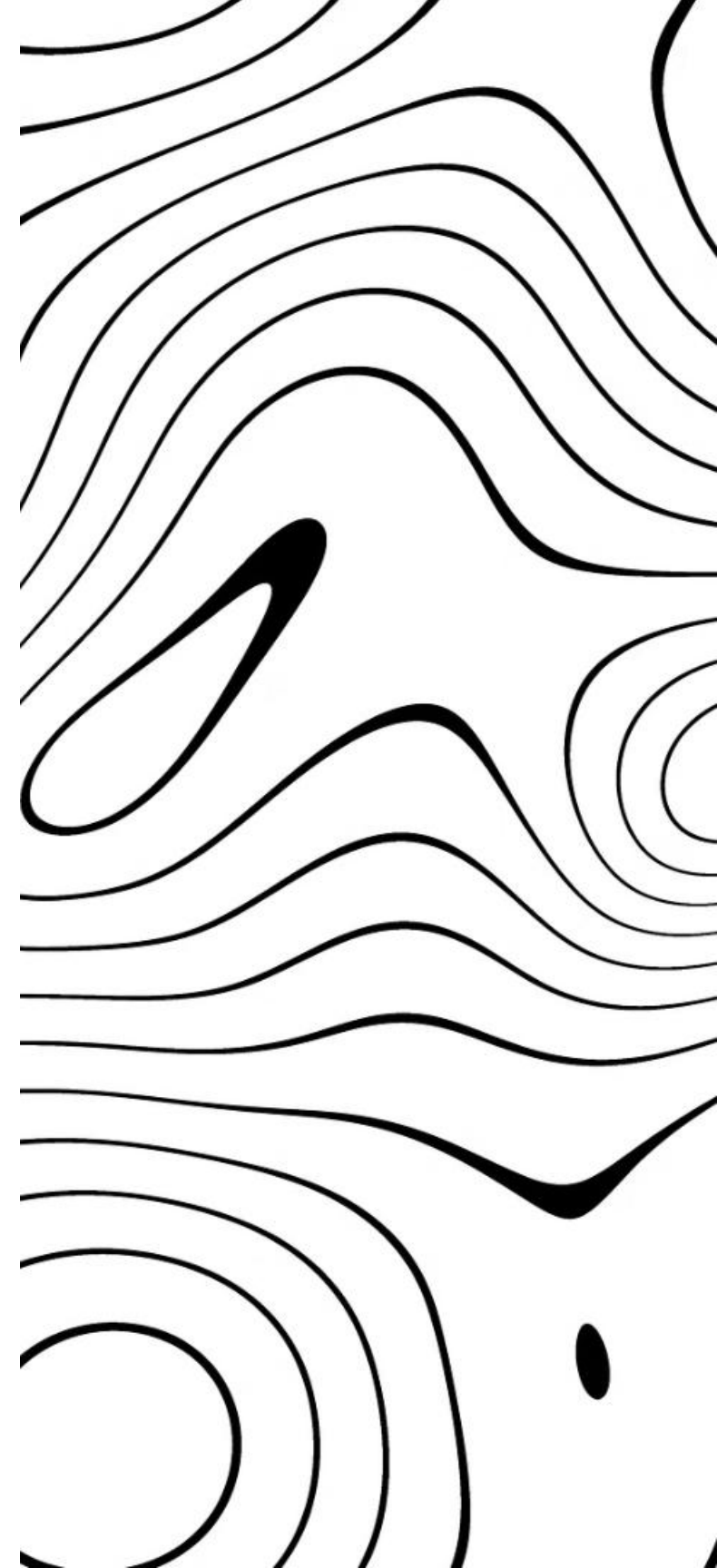
Privalumai

Daug greitesnė nei nuosekli paieška. Veikimo laikas yra $O(\log n)$.

Trūkumai

Reikalauja, kad masyvas būtų surūšiuotas. Tai riboja pritaikymą.

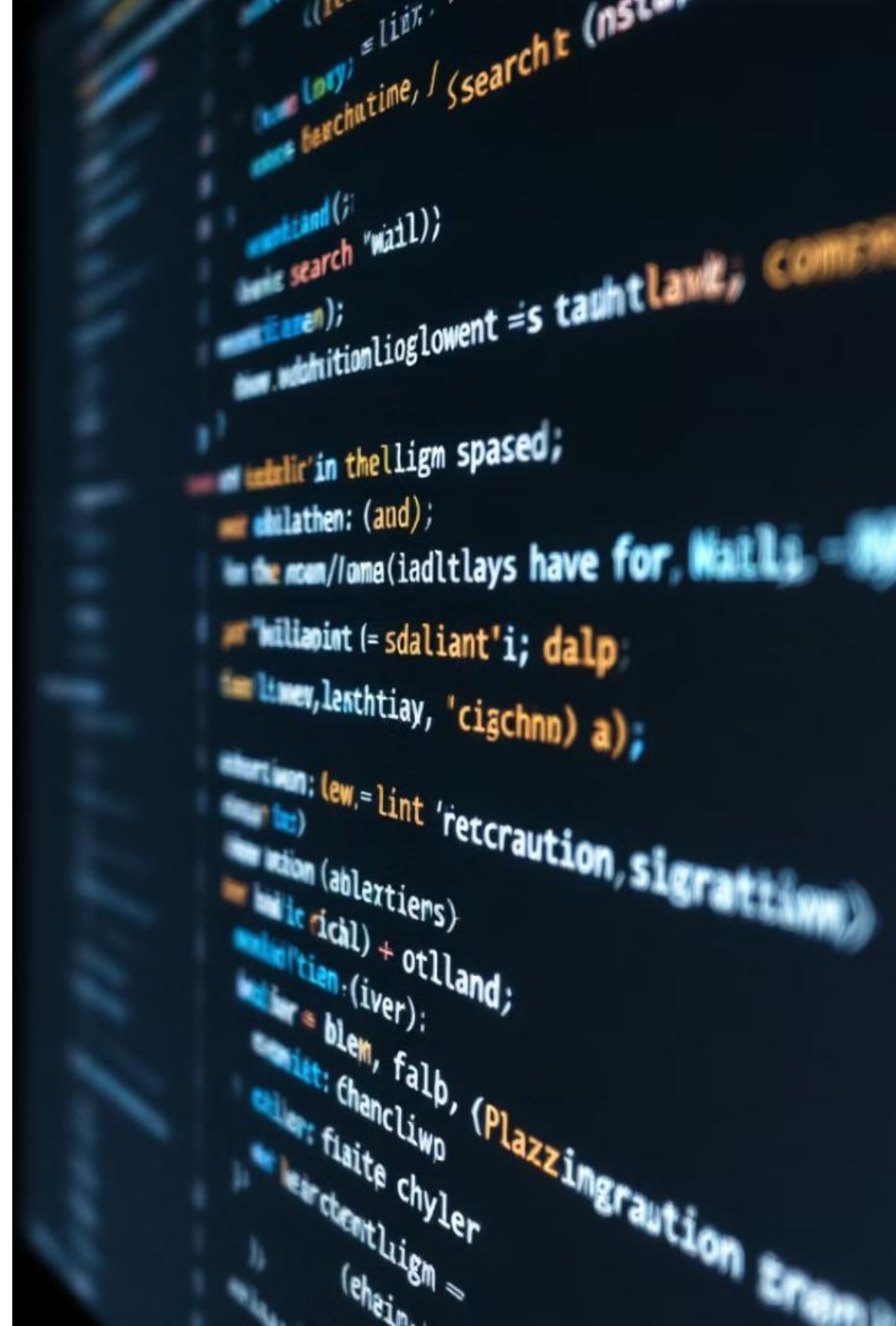
Dvejtainė paieška yra galingas, bet reikalaujantis tam tikros tvarkos metodas.



Dvejtainės Paieškos Pseudo Kodas

```
FUNKCIJA DvejetaiPaieska(masyvas, ieskomaReiksme):
    PRADZIA = 0
    PABAIGA = masyvas.ilgis - 1
    KOL PRADZIA <= PABAIGA:
        VIDURYS = (PRADZIA + PABAIGA) / 2
        JEI masyvas[VIDURYS] == ieskomaReiksme:
            GRAŽINTI VIDURYS
        JEI masyvas[VIDURYS] < ieskomaReiksme:
            PRADZIA = VIDURYS + 1
        KITAIP:
            PABAIGA = VIDURYS - 1
    GRAŽINTI -1 (elementas nerastas)
```

Atkreipkite dėmesį į masyvo dalijimą ir vidurio reikšmės tikrinimą. Tai esminiai dvejetainės paieškos žingsniai.





Praktinis Darbas: Algoritmų Palyginimas



Rašykite kodą

Igyvendinkite abu algoritmus pasirinkta programavimo kalba.



Testuokite

Naudokite skirtingus duomenų rinkinius abiejų algoritmų testavimui.



Analizuokite

Palyginkite algoritmų veikimo laiką ir efektyvumą.

Praktinis darbas padės jums geriau suprasti algoritmų skirtumus. Stebėkite veikimo laiką su skirtingais duomenimis.

Apibendrinimas ir Pagrindiniai Aspektai

1

Nuosekli paieška

Paprasta, bet lėta.

2

Dvejtainė paieška

Greita, bet reikalauja rūšiavimo.

3

Efektyvumo analizė

Svarbu pasirinkti tinkamą algoritmą.

Atsiminkite, kad tinkamo algoritmo pasirinkimas priklauso nuo duomenų rinkinio ir užduoties.