

The background is a dark navy blue. In the top-left corner, there are two overlapping parallelogram shapes: a blue one in front of a light green one. In the bottom-left corner, there is a circular inset showing a close-up of a circuit board with various electronic components. In the top-right corner, there is a faint, stylized graphic of a circuit board or a series of overlapping lines.

Searching Algorithms

Gunay Abbasova

Searching Algorithms

Linear Search

Binary Search

Jump Search

Interpolation Search

Exponential Search

Fibonacci Search

searching algorithm saxladıǵı hər hansı bir məlumat strukturundan elementi əldə etmək üçün nəzərdə tutulmuşdur. Axtarış əməliyyatının növündən asılı olaraq bu alqoritmlər 2 cür olur

1.Uninformed Search (Bilmədən axtarış)

2.Informed Search (Bilərək axtarış)



axtarış sistemlərinin növləri

axtarış sistemləri ardıcıl və interval olur

- Ardıcıl Axtarış: Burada siyahı və ya massiv ardıcıl olaraq keçilir və hər bir element yoxlanılır. Məsələn: Xətti Axtarış.
- Interval Axtarış: Bu alqoritmlər xüsusi olaraq çeşidlənmiş məlumat strukturlarında axtarış üçün nəzərdə tutulmuşdur. Bu tip axtarış alqoritmləri Xətti Axtarışdan qat-qat səmərəlidir, çünki onlar dəfələrlə axtarış strukturunun mərkəzini hədəf alır və axtarış sahəsini yarıya bölürlər.



uninformed search(Bilmədən axtarış)

1

axtarış prosesinin bilmədən aparılması bir sıra çətinliklərə səbəb olur məsələn problemə xas asanlıqları alqoritm bilmir.hər vəziyyətdə eyni şəkildə işləyən alqoritmlərə uninformed search (bilmədən axtarış) deyilir. uninformed search növləri:

1.Massivlər üzərində işləyənlər

2.Qraflar üzərində işləyənlər

3.Mətn Axtarış Alqoritmləri



Linear Search(xətti axtarış)

Linear search massivlər üzərində işləyən bilmədən axtarış növüdür. linear search ən təməl alqoritmlərdəndir . ən sadə axtarış sistemi linear search dur. Ancaq sürət baxımından nadir istifadə olunur burda massivdə elementi tapmaq üçün massivdəki hər bir elementi ardıcıl olaraq seçir və ardıcıl şəkildə yoxlayır . burada massivin solundan, 0ci indexindən başlayır, hər bir elementi bir bir X elementilə qarşılaşdırır. elementi tapana qədər davam edir. Linear searchdən massivlərdə element axtardıqda istifadə edirlər. Bu massivdə time complexity $O(n)$



Binary Search

- İkili axtarış, alqoritmin iki yarıya bölündüyü və elementin siyahının ortancil elementi ilə müqayisə edildiyi böl və fəth et metodu ilə işləyir. Evvelce massivin ortadaki elementi təyin olunur, axtarılan element ortancil elemental eyni olanda binary search sona çatır. Eyni olmayanda isə ortancil elementdən böyük olanda sağ tərəfə kiçik olanda sol tərəfə yeni kiçik elementə baxılır
- time complexity $O(1)$ -dir.
- Average Case Complexity - İkili axtarışın orta iş vaxtı mürəkkəbliyi $O(\log n)$
- Worst case complexity $O(\log n)$ -dir
- Worst case space complexity $O(1)$.



Jump Search

- İkili Axtarış kimi, Jump Search sort olunmuş massivlər üçün axtarış alqoritmidir., sabit addımlarla irəli tullanaraq və ya bütün elementləri axtarmaq əvəzinə bəzi elementləri atlayır. daha az elementi yoxlayır. Jump de time Xətti Axtarış ($O(n)$) və İkili Axtarış ($O(\log n)$) arasındadır.



Interpolation Search

Massivlər üzərində işləyən bilmədən axtarış növüdür. Sıralanmış massivlərə tətbiq olunur

Binary search üzərində daha təkmilləşdirmədir. time complexity burada $O(n)$ dir.

Bu axtarış alqoritmi tələb olunan dəyərin yoxlama mövqeyində işləyir. Bu alqoritmin düzgün işləməsi üçün məlumatlar düzgün paylanmış olmalıdır.



Exponential Search

- Eksponensial Axtarış, nisbətən mürəkkəbdir. həmçinin barmaq axtarışı kimi tanınır, sıralanmış massivlərə tətbiq olunur hər iterasiyada 2^i elementi atlayaraq çeşidlənmiş massivdə elementi axtarır, burada i dövrə nəzarət dəyişəninin dəyərini təmsil edir və sonra axtarış elementinin son atlama ilə cari keçid arasında mövcud olub-olmadığını yoxlayır. .
- Time complexity: $O(\log n)$
- Exponential searchdə massiv sort olunmuş olmalıdı və içərisində istənilən qədər element ola bilər.



Fibonacci Search

- Proqramlaşdırmada, Fibonacci ardıcılığı döngələr və rekursiya arasındakı fərqi müəyyən etməyə gəldikdə mürəkkəb yan görünüş təmin edir.
- Fibonaççi ardıcılığı 0-dan başlayan nömrələr qrupudur, burada növbəti nömrə son nömrənin + cari nömrənin əlavəsidir. Fibonacci alqoritminde time complexity $\log n$ -dir. Fibonacci alqoritmi çox sürətli işləyən bir alqoritmdir.