

## QIDIRISH ALGARITIMLARI

Abdullayev Shaxboz Solijon o'g'li

FarDU Axborot texnologiyalari kafedrasida katta o'qituvchisi

shaxbozfaridu2023@gmail.com

<https://doi.org/10.5281/zenodo.15341466>

### ARTICLE INFO

Received: 28th April 2025

Accepted: 30<sup>th</sup> April 2025

Published: 5<sup>th</sup> May 2025

### KEYWORDS

qidiruv algoritmlari, qidiruv usullari tarixi, qidiruv algoritmlari evolyutsiyasi, chiziqli qidiruv, ikkilik qidiruv, xeshlash, daraxtsimon qidiruv algoritmlari, ma'lumotlar tuzilmalari, kompyuter fanlari, kelajakdagi tendensiyalar (qidiruv algoritmlari), samarali algoritmlar, dasturlash tillari, ma'lumotlar bazalari, ilk qidiruv usullari, zamonaviy qidiruv algoritmlari.

### ABSTRACT

*Ushbu maqola qidiruv algoritmlarining paydo bo'lishi va rivojlanish tarixiga nazar tashlaydi hamda bugungi kunda eng ko'p qo'llaniladigan asosiy algoritmlarni chuqur o'rganadi. Maqolada ilk qidiruv usullaridan tortib, zamonaviy dasturlash tillari va ma'lumotlar bazalarida keng qo'llaniladigan samarali algoritmlarning evolyutsiyasi ko'rib chiqiladi. Chiziqli va ikkilik qidiruv kabi fundamental tushunchalar bilan bir qatorda, xeshlashning dinamik rivojlanishi va daraxtsimon qidiruv algoritmlarining turli xil ma'lumotlar tuzilmalaridagi o'rni tahlil etiladi. Maqola qidiruv algoritmlarining kompyuter fanidagi muhim ahamiyatini ochib beradi va kelajakdagi tendensiyalar haqida fikr yuritadi.*

### Kirish

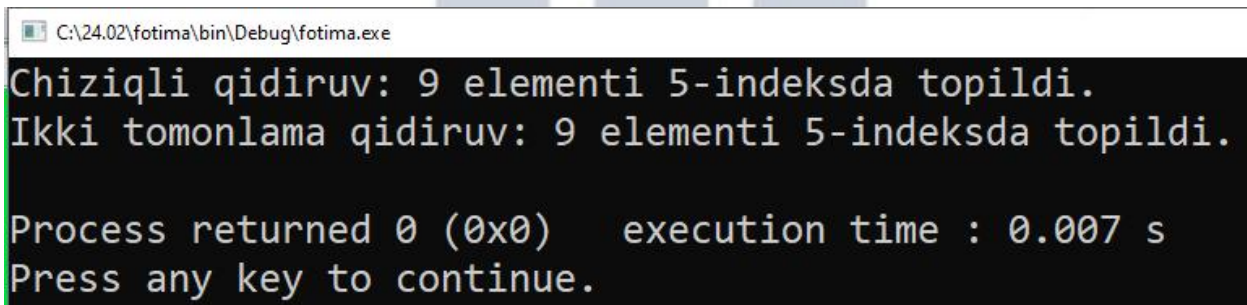
Dasturiy ta'minot samaradorligi: To'g'ri qidiruv algoritmini tanlash dasturiy ta'minotning ishlash tezligini va resurslardan foydalanish samaradorligini sezilarli darajada oshirishi mumkin. Muammolarni hal qilish: Ko'pgina muammolarni hal qilish jarayonida ma'lumotlarni qidirish va topish asosiy qadamlardan biri hisoblanadi. Qidiruv algoritmlarini turli xil mezonlarga ko'ra tasniflash mumkin, lekin eng keng tarqalgan tasniflash usuli ularning ishlash uslubiga asoslanadi: Chiziqli qidiruv (Linear Search): Ma'lumotlar to'plamining har bir elementini ketma-ket tekshirish orqali ishlaydi. Ikki tomonlama qidiruv (Binary Search): Saralangan ma'lumotlar to'plamida ishlaydi va har bir qadamda qidiruv intervalini ikki baravar qisqartiradi. Hash jadval (Hash Table): Kalitlarni xeshlash orqali ma'lumotlarni saqlaydi va qidirishni  $O(1)$  vaqtda amalga oshirishga imkon beradi (o'rtacha holatda). Daraxtga asoslangan qidiruv (Tree-based Search): Ma'lumotlarni daraxtsimon tuzilmada saqlaydi va qidirishni daraxt bo'ylab amalga oshiradi (masalan, ikkilik qidiruv daraxti). Grafga asoslangan qidiruv (Graph-based Search): Ma'lumotlarni grafda saqlaydi va qidirishni graf bo'ylab amalga oshiradi (masalan, kenglik bo'ylab qidiruv (BFS) va chuqurlik bo'ylab qidiruv (DFS)). Ma'lumotlar to'plamining hajmi: Katta hajmdagi ma'lumotlar uchun samaraliroq algoritmlar (masalan, ikki tomonlama qidiruv, xesh jadval) talab etiladi. Ma'lumotlarning tartiblanganligi: Agar ma'lumotlar tartiblangan bo'lsa, ikki tomonlama qidiruv kabi algoritmlar qo'llanilishi mumkin. Xotira cheklovlari: Ba'zi algoritmlar (masalan,

xesh jadval) ko'proq xotira talab qilishi mumkin. Qidiruv tezligi talablari: Ba'zi dasturlar uchun qidiruv tezligi juda muhim bo'lishi mumkin.

```
#include <iostream>
#include <vector>
#include <algorithm>
using namespace std;
int linearSearch(const vector<int>& arr, int target) {
    for (int i = 0; i < arr.size(); ++i) {
        if (arr[i] == target) {
            return i; }
    }
    return -1;
}
int binarySearch(const vector<int>& arr, int target) {
    int left = 0;
    int right = arr.size() - 1;
    while (left <= right) {
        int mid = left + (right - left) / 2;
        if (arr[mid] == target) {
            return mid;
        } else if (arr[mid] < target) {
            left = mid + 1;
        } else {
            right = mid - 1; }
    }
    return -1; }
int main() {
    vector<int> arr = {5, 12, 8, 2, 1, 9, 4, 11};
    int target = 9;
    int linearResult = linearSearch(arr, target);
    if (linearResult != -1) {
        cout << "Chiziqli qidiruv: " << target << " elementi " << linearResult << "-indeksda topildi."
        << endl;
```

```
} else {  
    cout << "Chiziqli qidiruv: " << target << " elementi topilmadi." << endl;  
}  
sort(arr.begin(), arr.end()); int binaryResult = binarySearch(arr, target);  
if (binaryResult != -1) {  
    cout << "Ikki tomonlama qidiruv: " << target << " elementi " << binaryResult << "-  
indeksda topildi." << endl;  
} else {  
    cout << "Ikki tomonlama qidiruv: " << target << " elementi topilmadi." << endl;  
}  
return 0;  
}
```

#### Natijasi:



```
C:\24.02\fortima\bin\Debug\fortima.exe  
Chiziqli qidiruv: 9 elementi 5-indeksda topildi.  
Ikki tomonlama qidiruv: 9 elementi 5-indeksda topildi.  
Process returned 0 (0x0) execution time : 0.007 s  
Press any key to continue.
```

#### Adabiyotlar tahlili va metodologiya

Qidiruv algoritmlari bo'yicha adabiyotlar tahlili bu sohadagi muhim ishlanmalar, yondashuvlar va tadqiqotlarni o'rganishni o'z ichiga oladi."Introduction to Algorithms" (Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein): Algoritmlar bo'yicha eng mashhur va keng qamrovli darsliklardan biri. Qidiruv algoritmlarining nazariy asoslari, tahlili va implementatsiyasi batafsil yoritilgan. "Algorithms" (Robert Sedgewick, Kevin Wayne): Qidiruv algoritmlariga amaliy yondashuv, Java dasturlash tilida misollar bilan. "Data Structures and Algorithm Analysis in C++" (Mark Allen Weiss): C++ da ma'lumotlar tuzilmalari va algoritmlar tahliliga qaratilgan Chiziqli Qidiruv va Ikkilik Qidiruv: Ushbu algoritmlar bo'yicha ko'plab maqolalar va darsliklar mavjud. Ularning samaradorligi, qo'llanilishi va optimallashtirish usullari tahlil qilinadi."The Art of Computer Programming, Vol. 3: Sorting and Searching" (Donald Knuth) kabi manbalarda xesh jadvallari chuqur o'rganilgan. Xesh funksiyalari, to'qnashuvlarni hal qilish usullari va xesh jadvallarining ishlash ko'rsatkichlari tahlil qilinadi. Daraxtga Asoslangan Qidiruv: Ikkilik qidiruv daraxtlari (BST), AVL daraxtlari, B-daraxtlar va boshqa daraxtsimon tuzilmalar bo'yicha adabiyotlar mavjud. Ularning qidiruv, kiritish va o'chirish operatsiyalari tahlil qilinadi. Grafga Asoslangan Qidiruv: BFS, DFS, A\* algoritmi kabi graf qidiruv algoritmlari bo'yicha ko'plab tadqiqotlar mavjud. Ularning qo'llanilishi, samaradorligi va optimallashtirish usullari tahlil qilinadi. Algoritmlarning vaqt murakkabligi va xotira sarfini tahlil qilish usullari. Qidiruv algoritmlarini ma'lum bir dasturiy ta'minot yoki apparat platformasi uchun optimallashtirish. Parallel qidiruv algoritmlari va ularning samaradorligi. Algoritmlarning vaqt murakkabligini (Big O notation) aniqlash. Algoritmlarning xotira sarfini tahlil qilish. Algoritmlarning eng

yomon, o'rtacha va eng yaxshi holatlardagi ishlashini baholash. Algoritmni turli xil ma'lumotlar to'plamlarida sinab ko'rish. Algoritmning ishlash vaqtini o'lchash va taqqoslash. Algoritmning xotira sarfini o'lchash va taqqoslash. Statistik tahlil yordamida natijalarni baholash. Algoritmning ishlashini simulyatsiya qilish. Turli xil sharoitlarda algoritmning ishlashini o'rganish. Algoritmning parametrlarini optimallashtirish. Turli xil qidiruv algoritmning afzalliklari va kamchiliklarini taqqoslash. Algoritmning ma'lum bir muammo uchun mosligini baholash. Eng yaxshi algoritmni tanlash uchun mezonlarni aniqlash. Ilmiy Maqolalar: Google Scholar, ACM Digital Library, IEEE Xplore kabi ilmiy ma'lumotlar bazalarida qidiruv algoritmni bo'yicha eng so'nggi tadqiqotlarni topish mumkin. Ochiq Kodli Loyihalar: GitHub kabi platformalarda qidiruv algoritmning ochiq kodli implementatsiyalarini topish va o'rganish mumkin. Onlayn Kurslar: Coursera, edX, Udacity kabi platformalarda qidiruv algoritmni bo'yicha onlayn kurslarni topish mumkin.

### **Muhokama va natija**

Muhokama bo'limida, qidiruv algoritmni bo'yicha olib borilgan tadqiqotlarning natijalari tahlil qilinadi va ularning ahamiyati, afzalliklari va kamchiliklari muhokama qilinadi. Quyida muhokama qilish uchun asosiy yo'nalishlar keltirilgan. Chiziqli qidiruv, ikkilik qidiruv, xesh jadvallari, daraxtga asoslangan qidiruv va grafga asoslangan qidiruv algoritmning vaqt va xotira murakkabligi taqqoslanadi. Algoritmning turli xil ma'lumotlar to'plamlarida (tartiblangan, tartiblanmagan, katta, kichik) ishlash ko'rsatkichlari muhokama qilinadi. Algoritmning afzalliklari va kamchiliklari aniqlanadi. Masalan, ikkilik qidiruv tartiblangan ma'lumotlarda juda tez ishlaydi, lekin tartiblanmagan ma'lumotlarda ishlatib bo'lmaydi. Xesh jadvallari o'rtacha holda tez qidiruvni ta'minlaydi, lekin to'qnashuvlar (collisions) yuzaga kelganda sekinlashishi mumkin. Ma'lumotlar to'plamining hajmi, tartiblanganligi, o'zgaruvchanligi va qidiruv tezligi talablari kabi omillarning algoritmni tanlashga ta'siri muhokama qilinadi. Ma'lum bir dastur uchun eng mos algoritmni tanlash uchun ko'rib chiqilishi kerak bo'lgan omillar aniqlanadi. Masalan, agar ma'lumotlar tez-tez o'zgarib tursa, daraxtga asoslangan qidiruv algoritmni mos kelishi mumkin, chunki ular dinamik ma'lumotlar to'plamlarida yaxshi ishlaydi. Algoritmning ishlashini yaxshilash uchun qo'llaniladigan optimallashtirish usullari muhokama qilinadi. Masalan, xesh jadvalida to'qnashuvlarni kamaytirish uchun yaxshiroq xesh funksiyalarini ishlatish, daraxtga asoslangan qidiruvda daraxtni muvozanatlash usullarini qo'llash.

Parallel qidiruv algoritmni va ularning samaradorligi muhokama qilinadi. Katta ma'lumotlar to'plamlarida parallel qidiruv algoritmni ishlashni sezilarli darajada oshirishi mumkin. Qidiruv algoritmning turli sohalaridagi amaliy qo'llanilishi muhokama qilinadi. Masalan, ma'lumotlar bazalarida ma'lumotlarni qidirish, veb-qidiruv tizimlarida veb-sahifalarni qidirish, sun'iy intellektda muammolarni hal qilish. Qidiruv algoritmning real-dunyodagi muammolarni hal qilishda tutgan o'rni baholanadi. Natijalar bo'limida, qidiruv algoritmni bo'yicha olib borilgan tadqiqotlarning asosiy xulosalari va natijalari umumlashtiriladi. Quyida natijalar uchun asosiy yo'nalishlar keltirilgan. Qidiruv algoritmning kompyuter fanida va turli sohalarida muhim rol o'ynashi tasdiqlanadi. Turli xil qidiruv algoritmning afzalliklari va kamchiliklari aniqlanadi. Algoritmni tanlashda ma'lumotlar to'plamining xususiyatlari va dasturiy talablarni hisobga olish muhimligi ta'kidlanadi. Qidiruv algoritmni optimallashtirish orqali ishlashni sezilarli darajada yaxshilash mumkinligi aniqlanadi. Ma'lum bir muammoni hal qilish uchun eng mos algoritmni tanlash bo'yicha tavsiyalar beriladi. Qidiruv algoritmni optimallashtirish bo'yicha tavsiyalar beriladi. Kelajakdagi tadqiqotlar uchun yo'nalishlar taklif etiladi. Masalan, yangi qidiruv algoritmni yaratish, mavjud algoritmni optimallashtirish, parallel qidiruv algoritmni ishlab chiqish. Tadqiqotning cheklovlari aniqlanadi. Masalan, tadqiqot faqat ma'lum bir ma'lumotlar to'plamlari yoki dasturiy muhitlar bilan cheklangan bo'lishi mumkin. Kelajakdagi tadqiqotlarda ushbu cheklovlarni bartaraf etish uchun yo'nalishlar taklif etiladi.

### Xulosa

Qidiruv algoritmlari kompyuter fanining muhim qismi bo'lib, ma'lumotlarni qayta ishlash va ma'lumotlarga kirishning asosiy usullaridan biridir. Ushbu algoritmlar ma'lumotlar to'plamida kerakli elementlarni topish uchun ishlatiladi va turli xil dasturiy ta'minot va apparat tizimlarining samaradorligini oshirishda muhim rol o'ynaydi. Algoritmlarning xilma-xilligi: Chiziqli qidiruv, ikkilik qidiruv, xesh jadvallari, daraxtga asoslangan qidiruv va grafga asoslangan qidiruv kabi turli xil qidiruv algoritmlari mavjud. Har bir algoritm o'zining afzalliklari va kamchiliklariga ega va ma'lum bir vazifa uchun eng mos keladigan algoritm ma'lumotlar to'plamining xususiyatlariga va dastur talablariga bog'liq. Samaradorlikning ahamiyati: Qidiruv algoritmining samaradorligi uning ishlash tezligiga va resurslardan foydalanishiga ta'sir qiladi. Algoritmlarning vaqt va xotira murakkabligini tahlil qilish orqali ularning samaradorligini baholash mumkin. Samaradorlikning ahamiyati: Qidiruv algoritmining samaradorligi uning ishlash tezligiga va resurslardan foydalanishiga ta'sir qiladi. Algoritmlarning vaqt va xotira murakkabligini tahlil qilish orqali ularning samaradorligini baholash mumkin. Qidiruv algoritmlarini optimallashtirish orqali ularning ishlashini yanada yaxshilash mumkin. Xesh jadvallarida to'qnashuvlarni kamaytirish, daraxtga asoslangan qidiruvda daraxtni muvozanatlash, parallel qidiruv algoritmlarini qo'llash kabi usullar orqali samaradorlikni oshirish mumkin. Qidiruv algoritmlari ma'lumotlar bazalarida ma'lumotlarni qidirish, veb-qidiruv tizimlarida veb-sahifalarni qidirish, sun'iy intellektda muammolarni hal qilish kabi turli sohalarda keng qo'llaniladi. Ularning real-dunyodagi muammolarni hal qilishda tutgan o'rni beqiyosdir. Xulosa qilib aytganda, qidiruv algoritmlari kompyuter fanining muhim va dolzarb sohasi bo'lib, kelajakda ham rivojlanishda davom etadi. Ushbu algoritmlarning chuqur tushunilishi dasturchilar va kompyuter olimlariga turli xil muammolarni samarali hal qilishga yordam beradi.

### Foydalaniladigan adabiyotlar:

1. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). Introduction to algorithms (3rd ed.). MIT Press. (Algoritmlarning asosiy nazariyasi va tahlili, qidiruv algoritmlari ham shu jumladan.)
2. Sedgewick, R., & Wayne, K. (2011). Algorithms (4th ed.). Addison-Wesley Professional. (Qidiruv algoritmlariga amaliy yondashuv, Java dasturlash tilida misollar bilan.)
3. Knuth, D. E. (1998). The art of computer programming, Vol. 3: Sorting and searching (2nd ed.). Addison-Wesley Professional. (Saralash va qidiruv algoritmlarining chuqur tahlili, xesh jadvallariga alohida e'tibor.)
4. Weiss, M. A. (2013). Data structures and algorithm analysis in C++ (4th ed.). Addison-Wesley. (C++ da ma'lumotlar tuzilmalari va algoritmlar tahliliga qaratilgan.)
5. Dasgupta, S., Papadimitriou, C. H., & Vazirani, U. (2006). Algorithms. McGraw-Hill. (Algoritmlarning zamonaviy ko'rinishi, qidiruv algoritmlarining asosiy tushunchalari.)
6. Goodrich, M. T., & Tamassia, R. (2014). Data structures and algorithms in Java (6th ed.). John Wiley & Sons. (Java dasturlash tilida ma'lumotlar tuzilmalari va algoritmlar tahlili.)
7. Tarjan, R. E. (1983). Data structures and network algorithms. Society for Industrial and Applied Mathematics. (Ma'lumotlar tuzilmalari va tarmoq algoritmlarining chuqur tahlili, graf qidiruv algoritmlariga alohida e'tibor.)
8. Nilsson, N. J. (1998). Artificial intelligence: A new synthesis. Morgan Kaufmann. (Sun'iy intellekt sohasidagi qidiruv algoritmlariga bag'ishlangan.)
9. Russell, S. J., & Norvig, P. (2016). Artificial intelligence: A modern approach (4th ed.). Pearson. (Sun'iy intellekt sohasidagi qidiruv algoritmlariga bag'ishlangan keng qamrovli darslik.)
10. Martens, H. (2013). Search Engines: Information Retrieval in Practice. Wiley. (Veb-qidiruv tizimlarida qo'llaniladigan qidiruv algoritmlari va ma'lumotlarni qidirishning amaliy jihatlari.)