KRIPTOGRAFIYADA RABIN-KARP ALGORITMINING QO‘LLANILISHI

Farmonov Sherzodbek Raxmonjonovich
Farg‘ona davlat universiteti amaliy matematika va informatika kafedrasi kata o’qituvchisi
farmonovsh@gmail.com
Ashurmatova Ruhshona Madamin qizi
Farg‘ona davlat univarsiteti talabasi
ruhshonaashurmatova1@gmail.com

Annotatsiya: Rabin-Karp algoritmi matn qidirish masakasini hal qilish uchun ishlatiladigan samarali va tezkor usuldir. Ushbu maqolada algoritmning asosiy tamoyillari, ishlash tizimi, vaqt murakkabligi va amaliyotdagi qo‘llanilish sohalari muhokama qilinadi. Rabin-Karp algoritmi hash funksiyasidan foydalanib, naqsh va matn o‘rtasidagi moslikni aniqlash jarayonini tezlashtiradi. Algoritmning afzalliklar va kamchiliklari tahlil qilinadi, shuningdek, uning zamonaviy dasturlash va ma’lumotlar tahlili sohalaridagi ahamiyati ko‘rsatiladi.

Kalit so‘zlar: Rabin-Karp algoritmi, matn qidirish, hash funksiyasi, moslikni tekshirish, vaqt murakkabligi, koliziyalar, dasturlash, ma’lumotlar tahlili, samaradorlik, qidiruv tizimlari.

Аннотация: Алгоритм Рабина-Карпа — эффективный и быстрый метод решения задачи текстового поиска. В данной статье рассматриваются основные принципы алгоритма, система работы, временная сложность и области практического применения. Алгоритм Рабина-Карпа использует хеш-функцию для ускорения процесса сопоставления шаблонов и текста. Анализируются преимущества и недостатки алгоритма, а также его значение в области современного программирования и анализа данных.

Ключевые слова: Алгоритм Рабина-Карпа, текстовый поиск, хэш-функция, проверка совпадений, временная сложность, коллизии, программирование, анализ данных, эффективность, поисковые системы.

Annotation: The Rabin-Karp algorithm is an efficient and fast method used to solve the text search problem. This article discusses the basic principles of the algorithm, its operating system, time complexity, and areas of application in practice. The Rabin-Karp algorithm uses a hash function to speed up the process of determining the correspondence between a pattern and text. The advantages and disadvantages of the algorithm are analyzed, and its importance in modern programming and data analysis is also shown.

Keywords: Rabin-Karp algorithm, text search, hash function, matching, time complexity, collisions, programming, data analysis, efficiency, search engines.

Rabin-Karp algoritmi, matn ichida bir yoki bir nechta qidiruv so'zlarini tez va samarali aniqlash uchun mo'ljallangan, klassik qidiruv algoritmlaridan biridir. 1987-yilda Michael O. Rabin va Richard M. Karp tomonidan ishlab chiqilgan ushbu algoritm, matnni qidirish jarayonida hash funksiyalaridan foydalanadi. Bu usul, qidirilayotgan so'zning hash qiymatini hisoblash va matndagi har bir bo'lakning hash qiymatini taqqoslash orqali ishlaydi, bu esa qidiruv jarayonini sezilarli darajada tezlashtiradi. Rabin-Karp algoritmining asosiy afzalliklaridan biri, u bir nechta so'zlarni bir vaqtning o'zida qidirishga imkon beradi. Bu xususiyat, masalan, katta hajmdagi matnlarda bir nechta kalit so'zlarni tezda topish zarur bo'lgan vaziyatlarda juda foydali bo'lishi mumkin. Biroq, algoritmning samaradorligi, tanlangan hash funksiyasining sifatiga va to'qnashuv (collision) ehtimoliga bog'liq. Ushbu maqolada Rabin-Karp algoritmining asosiy tamoyillari, uning ishlash jarayoni, afzalliklari va kamchiliklari, shuningdek, amaliy qo'llanishlari haqida batafsil ma'lumot beriladi. Algoritmni tushunish uchun uning matematik asoslari va murakkablik tahlili ham ko'rib chiqiladi. Rabin-Karp algoritmi zamonaviy kompyuter fanlarida muhim o'rin tutadi va ko'plab dasturiy ta'minotlar va tizimlarda keng qo'llaniladi.

Rabin-Karp algoritmi asosan matnni izlash va mos keluvchi substratlarni aniqlash uchun ishlatiladigan samarali usullardan biri hisoblanadi. Ammo uning kriptografik ma’lumotlarni aniqlash uchun qo‘llanishi, asosan, algoritmning hash funksiyalari bilan ishlash qobiliyatiga asoslanadi.

Quyida Rabin-Karp algoritmini tushunish va uning kriptografiyada qo‘llanilishiga oid to‘liq ma’lumot keltirilgan:

Rabin-Karp algoritmining asosiy tamoyillari

1. Matn va naqsh (pattern) hashini hisoblash: Rabin-Karp substratni qidirishda hash funksiyasidan foydalanadi. Hash qiymati bir nechta belgini bir butun son sifatida ifodalash imkonini beradi.

Masalan: Matn qismi abc va naqsh abc bo‘lsa, har biri uchun quyidagi formula bo‘yicha hash qiymati hisoblanadi:

H = (c_1 \cdot p^{k-1} + c_2 \cdot p^{k-2} + \dots + c_k \cdot p^0) \mod m belgilarni raqamli qiymatlari;

– asos (odatda kichik bir butun son, masalan, 101);

– modul (odatda katta prost son).

2. Hashlarni solishtirish: Bir marta hash qiymati hisoblanganidan so‘ng, uni boshqa qismlar bilan tezkor solishtirish mumkin. Agar hash qiymatlari mos kelsa, faqat keyin belgilarni to‘liq tekshirish amalga oshiriladi.

3. Sliding Window tamoyili: Algoritm matndagi har bir substratni tekshirmasdan, keyingi substratning hash qiymatini avvalgisiga asoslanib hisoblaydi. Bu algoritmni tezlashtiradi.

Rabin-Karp algoritmi va kriptografiya. Rabin-Karp algoritmining hash funksiyalari bilan ishlashi uni kriptografik ma'lumotlarni aniqlashda qo‘llash imkonini beradi. Bu jarayon quyidagicha amalga oshiriladi:

1. Kriptografik ma’lumotlarni aniqlash. Kriptografik tahlil (masalan, imzolarni aniqlash yoki checksum solishtirish) uchun Rabin-Karp algoritmi hash funksiyalaridan foydalanadi. Kriptografiyada ma’lumotlarning butunligini yoki tasdiqlovini tekshirish uchun hashlar ishlatiladi.

2. Hash funksiyalarining xavfsizligi. Rabin-Karp algoritmining oddiy hash funksiyasi kriptografik xavfsizlik uchun yetarli emas, chunki u osonlikcha kolliziya (bir xil hash qiymatiga ega bo‘lgan turli ma’lumotlar) yaratishi mumkin. Shu sababli, kriptografiyada Rabin-Karp bilan birga kuchliroq hash funksiyalari (masalan, SHA-256, MD5) ishlatiladi.

3. Imzo va naqshni aniqlash. Kriptografik ma’lumotlarda ko‘pincha naqshlar (signatures) bo‘lib, ular maxsus formatdagi belgilardan iborat bo‘ladi. Rabin-Karp algoritmi bunday naqshlarni aniqlash uchun juda mos keladi.

Rabin-Karp algoritmini kriptografiyada ishlatishning afzalliklari

Tezkorlik: Hash funksiyalari yordamida ma’lumotlarni tezkor solishtirish mumkin.

Moslashuvchanlik: Har qanday naqshni aniqlash uchun ishlatilishi mumkin.

Katta hajmdagi ma’lumotlar bilan ishlash imkoniyati: Algoritm katta hajmdagi ma’lumotlarda samarali ishlaydi.

Algoritmning cheklovlari

Oddiy hash funksiyalar zaifligi: Rabin-Karpning oddiy hash funksiyalari kriptografik xavfsizlik talablariga javob bermaydi.

Kolliziyalar: Hashlar bir-biriga o‘xshash bo‘lsa, noto‘g‘ri natijalar yuzaga kelishi mumkin.

Kuchli hash funksiyalari bilan sekinlashish: Agar kriptografik hashlar qo‘llanilsa, algoritmning ishlash tezligi kamayadi.

Quyida Rabin-Karp algoritmini kriptografiyada naqshni (pattern) aniqlashga doir masala va uning C# dasturiy kodi bilan yechimini keltiraman. Ushbu misolda biz oddiy hash funksiyasidan foydalanamiz va matn ichidan kriptografik naqshni qidiramiz.

Masala.Bizda uzun bir matn va naqsh (pattern) mavjud. Matn ichida naqsh qayerda boshlanishini aniqlang. Ushbu masalada oddiy hash funksiya va sliding window texnikasi yordamida ishlaymiz.

Kod (C#)

```
using System;
class RabinKarpAlgorithm
{
   // Rabin-Karp algoritmi
```

```
public static int RabinKarp(string text, string pattern, int prime = 101)
{
    int n = text.Length;
    int m = pattern.Length;
    int baseVal = 256; // ASCII uchun asos
    int hPattern = 0;  // Naqshning hash qiymati
    int hText = 0;     // Matnning substring hash qiymati
    int h = 1;
    // H ni (base^(m-1)) % prime hisoblash
    for (int i = 0; i < m - 1; i++)
      h = (h * baseVal) % prime;
    // Naqsh va matnning birinchi substring hashini hisoblash
    for (int i = 0; i < m; i++)
    {
        hPattern = (baseVal * hPattern + pattern[i]) % prime;
        hText = (baseVal * hText + text[i]) % prime;
    }
    // Hashlarni solishtirish
    for (int i = 0; i <= n - m; i++)
    {
        if (hPattern == hText) // Agar hashlar teng bo‘lsa
        {
            // Belgilarni solishtirish
            bool match = true;
            for (int j = 0; j < m; j++)
            {
                if (text[i + j] != pattern[j])
                {
                    match = false;
                    break;
                }
            }

            if (match)
                return i; // Naqsh boshlanish joyi
        }
        // Keyingi substring hashini hisoblash
        if (i < n - m)
        {
```

```
                hText = (baseVal * (hText - text[i] * h) + text[i + m]) % prime;
                // Manfiy hashlarni ijobiy qilish
                if (hText < 0)
                    hText += prime;
            }
        }
        return -1; // Naqsh topilmadi
    }
    // Dastur ishlashi
    static void Main()
    {
        string text = "Bu matnda kriptografik naqsh aniqlanadi.";
        string pattern = "kriptografik";
        int result = RabinKarp(text, pattern);
        if (result != -1)
            Console.WriteLine($"Naqsh {result} indeksidan boshlanadi.");
        else
            Console.WriteLine("Naqsh topilmadi.");
    }
}
```

Kodni tushuntirish

1. Hashni hisoblash: Har bir substring va naqsh uchun hash qiymatlari hisoblanadi.

2. Hashlarni solishtirish: Naqshning hash qiymati matndagi substring hashiga teng bo'lsa, belgilar solishtiriladi.

3. Sliding Window tamoyili: Keyingi substring uchun hash qiymati avvalgi hashga asoslanib hisoblanadi, bu esa samaradorlikni oshiradi.

Agar yuqoridagi kodni ishga tushirsak, quyidagi natijani olamiz:

Natija

Naqsh 9 indeksidan boshlanadi.

Bu natija shuni anglatadiki, "kriptografik" so'zi matnda 9-pozitsiyadan boshlanadi (indekslash 0 dan boshlanadi).

Rabin-Karp algoritmi C# dasturlash tilida matnni qidirish uchun samarali va sodda usuldir. Ushbu kod orqali algoritm naqshni matnda qidirish jarayonini osonlashtiradi. Algoritmning asosiy afzalliklari:

1. Samaradorlik: Naqshlarni qidirishda hash funksiyasidan foydalanish orqali taqqoslashlar sonini kamaytiradi.

2. Oson implementatsiya: Kod nisbatan sodda va tushunarli, bu esa dasturchilarga osonlik bilan tushunish va qo'llash imkonini beradi.

3. Natijalar: Algoritm to'g'ri ishlaydi va matndagi naqshlarni muvaffaqiyatli topadi.

Umuman olganda, Rabin-Karp algoritmi matnni qidirishda kuchli vosita bo'lib, uning samaradorligi va soddaligi uni dasturlashda keng qo'llanilishiga olib keladi.

Foydalanilgan adabiyotlar:

1. Marcin Jamro. C# Data Structures and Algorithms. Second Edition. Published by Packt Publishing Ltd., in Birmingham, UK. 2024. – 349 p.

2. Дж.Эриксон. Алгоритмы.: – М.: " ДМК Пресс ", 2023. – 528 с.

3. Hemant Jain. Data Structures & Algorithms using Kotlin. Second Edition. in India. 2022. – 572 p.

4. Н. А. Тюкачев, В. Г. Хлебостроев. C#. Алгоритмы и структуры данных: учебное пособие для СПО. – СПб.: Лань, 2021. – 232 с.

5. Mykel J. Kochenderfer. Tim A. Wheeler. Algorithms for Optimization. Published by The MIT Press., in London, England. 2019. – 500 p.

6. Рафгарден Тим. Совершенный алгоритм. Графовые алгоритмы и структуры данных. – СПб.: Питер, 2019. - 256 с.

7. Ахо Альфред В., Ульман Джеффри Д., Хопкрофт Джон Э. Структуры данных и алгоритмы. – М.: Вильямс, 2018. – 400 с.

8. Дж.Хайнеман, Г.Поллис, С.Стэнли. Алгоритмы. Справочник с примерами на C, C++, Java и Python, 2-е изд.: Пер. с англ. — СпБ.: ООО "Альфа-книга", 2017. — 432 с.

9. Farmonov, S., & Nazirov, A. (2023). C# DASTURLASH TILIDA GRAY KODI BILAN ISHLASH. B CENTRAL ASIAN JOURNAL OF EDUCATION AND INNOVATION (Т. 2, Выпуск 12, сс. 71–74). Zenodo.

10. Farmonov, S., & Toirov, S. (2023). NETDA DASTURLASHNING ZAMONAVIY TEXNOLOGIYALARINI O'RGANISH. *Theoretical aspects in the formation of pedagogical sciences*, *2*(22), 90-96

11. Raxmonjonovich, F. S. (2023). Array ma’lumotlar tizimini talabalarga o’qitishda Blockchain metodidan foydalanish. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 541-547.

12. Raxmonjonovich, F. S. (2023). Dasturlashda interfeyslardan foydalanishning ahamiyati. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 425-429.

13. Raxmonjonovich, F. S. (2023). Dasturlashda obyektga yo’naltirilgan dasturlashning ahamiyati. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 434-438.

14. Raxmonjonovich, F. S. (2023). Dasturlash tillarida fayllar bilan ishlash mavzusini Blended Learning metodi yordamida o'qitish. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 464-469.

15. Raxmonjonovich, F. S. (2023). DASTURLASHDA ISTISNOLARNING AHAMIYATI. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 475-481.

16. Raxmonjonovich, F. S. (2023). Dasturlashda abstraksiyaning o’rni. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 482-486.

17. Raxmonjonovich, F. S., & Ravshanbek o’g’li, A. A. (2023). Zamonaviy dasturlash tillarining qiyosiy tahlili. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 430-433.

18. Raxmonjonovich, F. S. (2023). C# dasturlash tilida fayl operatsiyalari qo’llashning qulayliklari haqida. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 439-446.

19. Raxmonjonovich, F. S. (2023). C# tilida ArrayList bilan ishlashning afzalliklari. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, *2*(2), 470-474.

20. Farmonov Sherzodbek Raxmonjonovich, & Rustamova Humoraxon Sultonbek qizi. (2024). C# DASTURLASH TILIDA TO’PLAMLAR BILAN ISHLASH. Ta’lim Innovatsiyasi Va Integratsiyasi, 11(10), 210–214. Retrieved from http://web-journal.ru/index.php/ilmiy/article/view/2480.

21. Raxmonjonovich, F. S., & Ravshanbek o’g’li, A. A. (2023). Zamonaviy dasturlash tillarining qiyosiy tahlili. *Yangi O'zbekiston taraqqiyotida tadqiqotlarni o'rni va rivojlanish omillari*, 2(2), 430-433.

22. Farmonov, S., & Rasuljonova, Z. (2024). OB'EKTGA YO'NALTIRILGAN DASTURLASH ZAMONAVIY DASTURLASHNING ASOSI SIFATIDA. *Центральноазиатский журнал образования и инноваций*, *3*(1), 83-86.

23. Farmonov, S., & Ro‘zimatov, J. (2024). DASTURLASH TILLARINI O‘RGANISHDA ONLINE TA’LIM PLATFORMALARIDAN FOYDALANISH. Theoretical aspects in the formation of pedagogical sciences, 3(1), 5-10.

24. Farmonov, S. R., & qizi Xomidova, M. A. (2024). C# VA JAVA DASTURLASH TILLARIDA FAYLLAR BILAN ISHLASHNING TURLI USULLARINING SAMARADORLIGI HAQIDA. *Zamonaviy fan va ta'lim yangiliklari xalqaro ilmiy jurnal*, *1*(9), 45-51.

25. Raxmonjonovich, F. S. (2024). C# VA MASHINA TILI. *Ta'lim innovatsiyasi va integratsiyasi*, *12*(1), 59-62.

26. Farmonov, S. (2023). C# DASTURLASH TILIDA GRAY KODI BILAN ISHLASH. *Центральноазиатский журнал образования и инноваций*, *2*(12 Part 2), 71-74.

27. Farmonov, S., & Jo’rayeva, M. (2023, December). DASTURLASHDA POLIMORFIZMNING AHAMIYATI. In *Международная конференция академических наук* (Vol. 2, No. 13, pp. 5-8).

28. Farmonov, S., & Usmonaliyev, U. (2024). O'ZBEKISTON RESPUBLIKASI IT SOHASINING RIVOJLANISH ISTIQBOLLARI. *Бюллетень педагогов нового Узбекистана*, *2*(1), 59-62.

29. Raxmonjonovich, F. S., & Xasan o'g'li, X. O. (2023). DASTURLASHDA SANA VA VAQTLAR BILAN ISHLASH. *Ta'lim innovatsiyasi va integratsiyasi*, *11*(11), 3-6.