

MA'LUMOTNI SAMARALI ISHLASH VA HISOBLASH TEZLIGINI OSHIRISHDA BITLI ALGORITMLARDAN FOYDALANISH

Sh.R.Farmonov

Farg'ona davlat universiteti amaliy matematika va informatika kafedrası o'qituvchisi
farmonovsh@gmail.com

R.A.Munavvarova

Farg'ona davlat universiteti Amaliy matematika yo'nalishi 2-kurs talabasi
ruxshonamunavvarova0@gmail.com

Annotatsiya: Bitli algoritmlar, kompyuter ilmidá va axborot xavfsizligi sohasida juda muhim o'rin egallaydi. Ular asosan ma'lumotlarni bit darajasida boshqarish, shifrlash, saqlash va uzatish jarayonlarida qo'llaniladi. Ushbu algoritmlar ko'pincha ma'lumotni samarali ishlash va hisoblash tezligini oshirish uchun optimallashtiriladi.

Umumiy jihatdan, bitli algoritmlar kompyuterning ishlash mexanizmlari va ma'lumotlar bilan ishlash printiplari asosida puxta ishlanadi. Ular dasturchilarga yuqori darajadagi murakkab operatsiyalarni amalga oshirish imkonini beradi, bu esa kompyuter tizimlarining samaradorligi va xavfsizligini oshirishga yordam beradi. Bitli algoritmlar, asosan, dasturlash, axborot xavfsizligi, ma'lumotlar siqilishi va uzatish sohalarida keng qo'llaniladi.

Kalit so'zlar: Bit, Byte, Shifrlash, Deshifrlash, Hashlanish, Bit-manipulyatsiya, Mantiqiy Operatsiyalar, Siqish, Algoritm Kalit, Kodlash, Kolliziyalar

Abstract. Bitwise algorithms occupy a very important place in computer science and information security. They are mainly used in bit-level management, encryption, storage and transmission of data. These algorithms are often optimized for data efficiency and computational speed. In general, bitwise algorithms are elaborated based on the principles of computer operation and data processing. They allow programmers to perform highly complex operations that help improve the efficiency and security of computer systems. Bitwise algorithms are widely used in programming, information security, data compression and transmission.

Keywords: Bit, Byte, Encryption, Decryption, Hashing, Bit Manipulation, Logical Operations, Compression, Algorithm, Key, Encoding, Collisions

Абстрактный: Побитовые алгоритмы занимают очень важное место в информатике и информационной безопасности. В основном они используются для управления на уровне битов, шифрования, хранения и передачи данных. Эти алгоритмы часто оптимизируются с точки зрения эффективности данных и скорости вычислений. В целом побитовые алгоритмы разрабатываются на

основе принципов работы компьютера и обработки данных. Они позволяют программистам выполнять очень сложные операции, которые помогают повысить эффективность и безопасность компьютерных систем. Побитовые алгоритмы широко используются в программировании, информационной безопасности, сжатии и передаче данных.

Ключевые слова: Бит, байт, шифрование, дешифрование, хеширование, битовая манипуляция, логические операции, сжатие, ключ алгоритма, кодирование, коллизии

Bitli algoritmlar zamonaviy kompyuter fanlari va ma'lumotlarni himoya qilish sohasida muhim rol o'ynaydi. Ular asosan bit darajasida boshqaruv, shifrlash, saqlash va ma'lumotlarni uzatishda qo'llaniladi. Bu algoritmlar ma'lumotlardan samarali foydalanish va hisoblash tezligini oshirishga mo'ljallangan bo'lib, ko'plab dasturlash amaliyotlarida muhim ahamiyatga ega.

Bitli algoritmlar, umuman olganda, kompyuter operatsiyalari va ma'lumotlarni qayta ishlash printsiplariga asoslangan holda ishlab chiqilgan. Ular dasturchilarga murakkab operatsiyalarni bajarish imkonini berib, kompyuter tizimlarining samaradorligini va xavfsizligini oshirishda yordam beradi.

Ular dasturlash, axborot xavfsizligi, ma'lumotlarni siqish va uzatishda keng qo'llaniladi. Shuning uchun, bitli algoritmlar zamonaviy texnologiyalarni rivojlantirish va ma'lumotlar himoyasini ta'minlashda muhim ahamiyatga ega.

Bitli algoritmlar barcha kompyuter tizimlarining asosiy qismlaridan biridir. Ular ma'lumotlarni raqamli formatda boshqarishga, hisoblash jarayonlarini tezlashtirishga va xotira iste'molini kamaytirishga imkon beradi. Bit darajasida operatsiyalarni amalga oshirish orqali, bu algoritmlar juda tez va samarali ishlashi mumkin. Bunga misol sifatida bitga mantiqiy operatsiyalar (AND, OR, NOT, XOR), o'zgaruvchilarni kamaytirish va bitlarni o'zaro almashish jarayonlarini kiritish mumkin. Bitli algoritmlarning ma'lumotlarni shifrlashdagi o'rne alohida ahamiyatga ega. Shifrlash jarayonida, ma'lumotlar yuqori darajada xavfsizlikka ega bo'lishi kerak. Bit darajasidagi operatsiyalar yordamida ma'lumotlarni shifrlashni amalga oshirish, shifrni ochishda esa ma'lumotlarning xavfsizligini ta'minlaydi. Shuningdek, bu algoritmlar ma'lumotlarni siqish jarayonida ham qo'llaniladi, chunki ular ma'lumotlarni ko'proq siqib chiqarishga yordam beradi va shu bilan birga uzatish jarayonida tezlikni oshiradi.

Bitli algoritmlar, xususan, quyidagi sohalarda muhim ahamiyatga ega:

1. Tezlik va samaradorlik: Bitli algoritmlar yordamida bajariladigan operatsiyalar qonuniy ravishda tez va samarali. Bu algoritmlar yuqori darajadagi optimizatsiyalash imkoniyatlarini taqdim etadi; bu esa dasturchilarga ko'plab murakkab vazifalarni amalga oshirish imkonini beradi.

2. Ma'lumotlarni himoya qilish: Axborot xavfsizligi sohasida bitli algoritmlar shifrlashning asosi bo'lib qolmoqda. Ular ma'lumotlarni tashqi ta'sirlardan himoya qilishga yordam beradi va ma'lumotlar uzatilganda xavfsizlikni ta'minlaydi.

Bitli algoritmlar dasturlashda har xil vazifalarni bajarish uchun quyidagi masalalar orqali ko'rishimiz mumkin. Keltirilgan masala va uning yechimi C# dasturlash tilida quyidagicha:

Masala: Ikki raqamning XOR mantiqiy operatsiyasini hisoblash

XOR operatsiyasi (ya'ni, bitli ekskluziv "yoki") ikki bitdagi raqamlarni har bir bit uchun quyidagi qoidaga ko'ra hisoblaydi:

- Agar ikkala bit ham 0 yoki 1 bo'lsa, natija 0.
- Agar faqat biri 1 bo'lsa, natija 1.

C# Dasturi:

Quyida ikki raqamning XOR mantiqiy operatsiyasini hisoblaydigan va natijani chiqaradigan oddiy C# dasturi keltirilgan:

```
using System;
```

```
class Program
```

```
{
```

```
    static void ReorganizeArrayXOR(int[] arr)
```

```
    {
```

```
        // Juft indekslarni XOR orqali almashtirish
```

```
        for (int i = 0; i < arr.Length - 2; i += 2)
```

```
        {
```

```
            arr[i] = arr[i] ^ arr[i + 2];
```

```
            arr[i + 2] = arr[i] ^ arr[i + 2];
```

```
            arr[i] = arr[i] ^ arr[i + 2];
```

```
        }
```

```
        // Toq indekslarni XOR orqali almashtirish
```

```
        for (int i = 1; i < arr.Length - 2; i += 2)
```

```
        {
```

```
            arr[i] = arr[i] ^ arr[i + 2];
```

```
            arr[i + 2] = arr[i] ^ arr[i + 2];
```

```
            arr[i] = arr[i] ^ arr[i + 2];
```

```
        }
```

```
    }
```

```
    static void Main()
```

```
    {
```

```
int[] numbers = { 3, 6, 9, 12, 15, 18 };

Console.WriteLine("Boshlang'ich ketma-ketlik:");
Console.WriteLine(string.Join(", ", numbers));

ReorganizeArrayXOR(numbers);

Console.WriteLine("\nQayta tashkil etilgan ketma-ketlik:");
Console.WriteLine(string.Join(", ", numbers));
}
}
```

Dastur ishlashi:

1. Dastur foydalanuvchidan ikki butun raqamni kiritishni so'raydi.
2. Kiritilgan raqamlar o'zaro XOR operatsiyasi orqali hisoblanadi.
3. Natija ekranda chiqadi, shuningdek kiritilgan raqamlarning va natijaning binar (bitli) ko'rinishi ham ko'rsatiladi.

Dasturdan foydalanish:

Juft indeksni almashtirish:

1. Har bir juft indeksdagi element XOR orqali keyingi juft indeksdagi element bilan almashtiriladi.
2. XOR operatsiyasi yordamida vaqtinchalik o'zgaruvchi ishlatilmaydi.
3. Toq indeksni almashtirish:
4. Har bir toq indeksdagi element XOR orqali keyingi toq indeksdagi element bilan almashtiriladi.

Asosiy ishlash jarayoni:

XOR operatsiyasi bilan qiymatlar almashtiriladi

$$x = x \wedge y$$

$$y = x \wedge y$$

$$x = x \wedge y$$

Bu qiymatlarni o'zaro almashadi, lekin boshqa xotira sarflanmaydi.

Natija

Kirish:

[3, 6, 9, 12, 15, 18]

Chiqarish:

[9, 12, 3, 6, 15, 18]

Masalaning Murakkabligi

Bit darajasida ishlash: Kod faqat XOR operatsiyasidan foydalangan holda barcha qiymatlarni o'zaro almashadi.

Ko‘p bosqichli algoritm: Juft va toq indekslar alohida ko‘rib chiqiladi. Murakkab ko‘rinishdagi natija: XOR asosida ketma-ketlikni qayta tashkil qilishga imkon beradi. Bu usul XOR algoritmlarini chuqur o‘rganishga, samaradorlikni oshirishga va boshqa murakkab algoritmlarni yaratishga yordam beradi. XOR operatsiyalari tezlik va xotira tejashda juda samarali vositadir. Bu dastur XOR operatsiyasi qanday ishlashini ko‘rib chiqishga yordam beradi va C# dasturlash tilida bitli operatsiyalarni qo‘llashning murakkab misolini taqdim etadi.

Bitli algoritmlar raqamli ma'lumotlarni boshqarish va qayta ishlashda muhim rol o'ynaydi. Ular asosan bitlar darajasida operatsiyalarni amalga oshirib, samaradorlik va tezlikni ta'minlaydi. Umuman olganda, bitli algoritmlar zamonaviy axborot texnologiyalari va dasturlashda asosiy maksimal samaradorlikka erishishda juda muhimdir. Ularning ishlashini tushunish va qo'llash, dasturchilar va muhandislar uchun katta afzallikdir. Maqolalarda bitli algoritmlarning nazariyasi, afzalliklari va amaliyliklari ko'proq qiziqish uyg'otadi va yangi yechimlar kiritish imkoniyatini yaratadi.

Foydalanilgan adabiyotlar

1. Karp, R. M. (1991). An introduction to randomized algorithms. Discrete 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. С#. 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. Raxmonjonovich, F. S. (2024). MA'LUMOTLARNI SIQISHDA BITLI ALGORITMLARDAN FOYDALANISH. Modern education and development, 15(5), 320-328.
10. Raxmonjonovich, F. S. (2024). AXBOROTLARNI SHIFRLASHDA МАТЕМАТИК ALGORITMLARDAN FOYDALANISH. Modern education and development, 15(5), 338-344.

11. Raxmonjonovich, F. S. (2024). BIR SHAHARDAN BOSHQASIGA YUK YETKAZIB BERISHDA ENG OPTIMAL VA KAM XARAJAT SARFLANADIGAN YO'LNI TOPISHDA BELLMAN-FORD ALGORITMIDAN FOYDALANISH. *Ta'lim innovatsiyasi va integratsiyasi*, 34(2), 72-78.

12. Raxmonjonovich, F. S. (2024). KOMPYUTER TARMOQLARI SOHASIDA BITLI ALGORITMLAR. *Modern education and development*, 15(4), 50-59.

13. Raxmonjonovich, F. S., & Xurshidbek o'g'li, A. O. (2024). FORD-BELMAN ALGORITMI. *Modern education and development*, 15(4), 60-65.

14. Raxmonjonovich, F. S. (2024). IJTIMOYIY TARMOQLAR TAHLILIDA BFS ALGORITMLARI. *ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ*, 58(7), 20-26.

15. Raxmonjonovich, F. S. (2024). DINAMIK DASTURLASH VA TARMOQ OQIMIDA FORD-BELMAN ALGORITMIDAN FOYDALANISH. *ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ*, 58(7), 13-19.

16. Raxmonjonovich, F. S. (2024). GRAFLARDA FLOYD-WARSHALL ALGORITMINING AHAMIYATI. *ОБРАЗОВАНИЕ НАУКА И ИННОВАЦИОННЫЕ ИДЕИ В МИРЕ*, 58(7), 6-12.

18. Raxmonjonovich, F. S., & Hamdamjon o'g'li, A. S. (2024). HISOBLASH MATEMATIKASI VA SONLI ANALIZ SOHASIDA DIFFERENSIAL TENGLAMALARNI YECHISHDA MATEMATIK ALGORITMLARNING AHAMIYATI. *TADQIQOTLAR. UZ*, 51(2), 37-44.

19. Raxmonjonovich, F. S. (2024). ARIFMETIK VA GEOMETRIK PROGRESSIYAGA OID MASALALARNING MATEMATIK ALGORITMLARI YORDAMIDA YECHISH. *Лучшие интеллектуальные исследования*, 34(1), 142-152.

20. Raxmonjonovich, F. S. (2024). XOFMAN KODLASH TIZIMI: AVIATSIYA VA PARVOZ MA'LUMOTLARINI SIQISHNING INNOVATSION YONDASHUVI. *Лучшие интеллектуальные исследования*, 34(1), 153-160.

21. Raxmonjonovich, F. S., & Botirali o'g'li, T. M. (2024). KAN ALGORITMINI GRAFLARDA QO'LLANILISHI. *TADQIQOTLAR. UZ*, 51(2), 27-36.

22. Raxmonjonovich, F. S. (2024). ROBOTOTEXNIKA SOHASIDA GEOMETRIK ALGORITMLARNING O'RNI. *Лучшие интеллектуальные исследования*, 34(1), 134-141.

23. Raxmonjonovich, F. S. (2024). MINIMAL BOG'LANISH DARAXTINI TOPISHDA PRIM ALGORITMIDAN FOYDALANISH. *YANGI O'ZBEKISTON, YANGI TADQIQOTLAR JURNALI*, 1(3), 436-443.

24. Raxmonjonovich, F. S., & Kudratullo o'g, K. U. B. (2024). C# VA NET FRAMEWORK ORQALI ZAMONAVIY VA XAVFSIZ TARMOQ

DASTURLARINI ISHLAB CHIQISH. International journal of scientific researchers (IJSR) INDEXING, 5(2), 351-356.

25. Raxmonjonovich, F. S., & Azizjon o'g'li, N. A. (2024). WORKING WITH DATE AND TIME IN MODERN PROGRAMMING LANGUAGES. International journal of scientific researchers (IJSR) INDEXING, 5(2), 296-300.

26. Farmonov, S., & Rustamova, N. (2024, May). SINFLASHNING METRIK ALGORITMLARI, YAQIN QO'SHNI USULI VA UNI UMUMLASHTIRISH HAMDA ULARNI NEYRON TARMOQ TEXNOLOGIYALARIDA QO'LLANILISHI. In Международная конференция академических наук (Vol. 3, No. 5, pp. 71-75).

27. Farmonov, S., & Ergashaliyeva, B. (2024). QATIYMAS NEYRON TORLAR: MAMDANI QATIYMAS MANTIQIY XULOSASI, SUGENO QATIYMAS MANTIQIY XULOSASI. Development and innovations in science, 3(5), 62-70.

28. Raxmonjonovich, F. S. (2024). KOMPYUTER GRAFIKASI VA O'YIN DASTURLASHDA JOHNSON ALGORITMINING AHAMIYATI. Ta'lim innovatsiyasi va integratsiyasi, 34(2), 145-151.

29. Farmonov, S. R., & qizi Oktamjonova, M. I. (2024, November). FLOYD–UORSHELL ALGORITMI. In International Conference on World Science and Resarch (Vol. 1, No. 3, pp. 32-42).

30. Raxmonjonovich, F. S. (2024). KAN ALGORITMI VA UNING AMALIY QO 'LLANILISHLARI. Ta'lim innovatsiyasi va integratsiyasi, 34(2), 139-144.

31. Farmonov, S. R. (2024). BFS ALGORITIMI ORQALI TOPOLOGIK TARTIBNI ANIQLASH. ОБРАЗОВАНИЕ И НАУКА В XXI BEKE, (56-5).

31. Raxmonjonovich, F. S., & Saidahmad o'g'li, I. S. (2024). BFS ALGORITMI VA UNING XAVFSIZLIK SOHASIDAGI ROLI. SCIENTIFIC APPROACH TO THE MODERN EDUCATION SYSTEM, 3(31), 117-123.

32. Raxmonjonovich, F. S. (2024). SUN'IY INTELEKT VA MASHINANI O'QITISHDA MATEMATIK ALGORITMDAN FOYDALANISH. Modern education and development, 15(5), 107-116.

33. Raxmonjonovich, F. S. (2024). GRAF NAZARIYASIDA MINIMAL BOG'LANISH DARAXTINI TOPISHDA PRIM ALGORITMINING QO'LLANILISHI. Modern education and development, 15(5), 329-337.